AFRL-IF-RS-TR-2005-378
Final Technical Report
November 2005

ADVERSARIAL INTENT INFERENCE FOR PREDICTIVE BATTLESPACE AWARENESS

University of Connecticut

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-378 has been reviewed and is approved for publication

APPROVED: /s/
JAMES HANNA
Project Engineer

FOR THE DIRECTOR: /s/
JAMES W. CUSACK, Chief
Information Systems Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 2005	**Final** **Sep 01 – Apr 05**

4. TITLE AND SUBTITLE

ADVERSARIAL INTENT INFERENCE FOR PREDICTIVE BATTLESPACE AWARENESS

5. FUNDING NUMBERS

G - F30602-01-1-0595
PE - 62702F
PR - 558B
TA - II
WU - 09

6. AUTHOR(S)

Eugene Santos, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Connecticut
Office for Sponsored Programs
438 Whitney Road, Extn, Unit 1133
Storrs CT 06269-2938

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/IFTC
525 Brooks Road
Rome NY 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TR-2005-378

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: James Hanna/IFTC/(315) 330-3473 James.Hanna@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 Words)***

Designed and developed a cognitive architecture to model the adversary which forms the basis for the Adversary Intent Inferencing (AII) Module. Designed, developed, and implemented the AII Module based on both Bayesian Networks and Bayesian Knowledge Bases for adversarial modeling, course of action prediction, explanation, and inference of adversary intent. All functioning in both wintel and Unix environments. Integrated AII module into prototype system for modeling and predicting the adversary based on the Battle of Khafji scenario. Integrated AII module in Force Structure Simulation wargaming system and demonstrated emergent behavior when Red force behavior is varied (AFRL/IFTC). Developed Dynamico tool for constructing Bayesian fragments and templates into libraries for use by the AII – resulted in a MS Thesis. AII transitioned into Phase I and Phase II SBIR Projects under Securboration – "Emergent Adversarial Modeling System".

14. SUBJECT TERMS

Adversary Intent Inferencing Module, Bayesian Networks, Bayesian Knowledge Bases, Force Structure Simulation

15. NUMBER OF PAGES 77

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

1.0 Final Project Summary

Goal: Design and develop computational framework for adversarial modeling and intent inferencing for decision support

Approach: Dynamically capture and predict enemy interests, goals, rationale, and courses of action under uncertainty through machine learning and Bayesian networks

To achieve adversarial intent inferencing requires the ability to (1) fuse information (observables) from sensors and intelligence sources regarding the adversary, (2) infer adversary intent and goals, and (3) predict adversary courses of action (COA). In total, adversary intent inferencing (AII) provides these three key functions while also taking into consideration a number of utility issues:

- AII must be able to explain the basis of its predictions; why is the adversary pursuing a predicted goal? What is driving the adversary to pursue these COAs? Must be able to model and take into account many factors including soft factors such as political environment, personality issues, adversarial religious beliefs, etc.
- AII must be able to adapt predictions based on history of events and observed enemy operations.
- AII must be dynamic and able to learn changes in the adversary behavior and ultimately model pop-up adversaries; AII must be able to provide the capability for standing up models of new adversaries while avoiding the knowledge/information engineering bottleneck.

While the ultimate role and capabilities of AII still requires a great deal of long term research, from our first steps thus far, we can carefully identify capabilities that are likely achievable in the short-term for mission planning and wargaming to support Effects-based Operations (EBO), Predictive Battlespace Awareness (PBA), and Intelligent Preparation of the Battlespace (IPB).

Lockheed Martin Advanced Technology Labs (LM ATL) in Cherry Hill, NJ were subcontracted under this grant to help provide prototyping and proof-of-concept systems development.

2.0 Major Accomplishments

- Designed and developed a cognitive architecture to model the adversary which forms the basis for the Adversary Intent Inferencing (AII) Module.

- Designed, developed, and implemented the Adversary Intent Inferencing (AII) Module based on both Bayesian Networks and Bayesian Knowledge Bases for adversarial modeling, course of action prediction, explanation, and inference of adversary intent. AII functioning in both wintel and Unix environments.
- AII module integrated into prototype system for modeling and predicting the adversary based on the Battle of Khafji scenario.
- AII module integrated in Force Structure Simulation wargaming system and demonstrated emergent behavior when Red force behavior is varied (AFRL/IFTC).

- Developed Dynamico tool for constructing Bayesian fragments and templates into libraries for use by the AII – resulted in a MS Thesis.
- AII transitioned into Phase I and Phase II SBIR Project under Securboration – "Emergent Adversarial Modelling System."
-

3.0 AFRL/IF Project Conclusion and Future Work

Conclusion

Through our prototype of the Battle of al Khafji during the First Persian Gulf War with LM ATL, we have demonstrated that we can successfully perform adversarial intent inferencing that can fuse observables from intelligence sources, infer adversary intentions and goals, and predict adversary actions. In particular, our simulation of the Battle of al Khafji included the stream of enemy observables as they unfolded in the battlefield. The initial intent of the adversary was not to attack across the Saudi border into al Khafji as was believe by military intelligence and leaders at the time. As the situation unfolded in the simulation, the adversary model evolved the underlying intent dynamically based on the observables and predicted enemy actions in accordance with the actions taken during the battle. This demonstrated that our cognitive architecture could capture adversary intent and change over time which also showed the dynamics inherent in such situations.

We also successfully demonstrated that our probabilistic network modeling approach to be viable to capturing the scenario and provided guidelines on how to construct such models through a tool we developed, Dynamico, that helped construct adversary network templates and fragments.

Future Work

We have begun transition of the adversary intent inferencing architecture into an AFRL/IF Phase I and II SBIR entitled "Emergent Adversary Modeling System" with Securboration, Inc. The goal of this work is to fully move the model into an environment that can be integrated within modern wargaming and mission planning systems.

Among the future work for adversary intent inferencing is to develop a formal framework for automatically building adversaries using Bayesian Knowledge Fragments and incorporating a psychologically and cognitively validated model to account for cultural, political, social, and economic factors influencing the adversary. In addition, more in depth scenarios should be explored as well as modeling groups of adversaries and the effect of inter- and intra-group relationships.

4.0 Appendix A. Publications

[1 book chapter, 6 conference publications, and 1 MS Thesis]
[The publications below were supported in full or in part by this project. Papers 2-7 are attached to the end of this report.]

1. Santos, Eugene, Jr. and Zhao, Qunhua, "Adversarial Models for Opponent Intent Inferencing," to appear in *Adversarial Reasoning* (Eds. A. Kott and W. McEneaney), CRC Press.
2. Bell, Benjamin, Santos, Eugene, Jr., and Brown, Scott M., "Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion," *Proceedings of the 11th Conference on Computer Generated Forces and Behavioral Representation*, 535-542, Orlando, FL, 2002.
3. Brown, Scott M., Santos, Eugene, Jr., and Bell, Benjamin, "Knowledge Acquisition for Adversary Course of Action Prediction Models," *Proceedings of the AAAI 2002 Fall Symposium on Intent Inference for Users, Teams, and Adversaries*, Boston, MA, 2002.
4. Surman, Joshua, Hillman, Robert, and Santos, Eugene, Jr., "Adversarial Inferencing for Generating Dynamic Adversary Behavior," *Proceedings of the SPIE: 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, Vol. 5091, 194-201, Orlando, FL, 2003.
5. Santos, Eugene, Jr., "A Cognitive Architecture for Adversary Intent Inferencing: Knowledge Structure and Computation," *Proceedings of the SPIE: 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, Vol. 5091, 182-193, Orlando, FL, 2003.
6. Revello, Timothy, McCartney, Robert, and Santos, Eugene, Jr., "Multiple Strategy Generation for War Gaming," *Proceedings of the SPIE: Defense & Security Symposium*, Vol. 5423, 232-243, Orlando, FL 2004.
7. Santos, Eugene, Jr. and Negri, Allesandro, "Constructing Adversarial Models for Threat Intent Prediction and Inferencing," *Proceedings of the SPIE Defense & Security Symposium*, Vol. 5423, 77-88, Orlando, FL 2004.
8. Negri, Allesandro, "Bayesian Templates," MS Thesis, Department of Computer Science & Engineering, University of Connecticut, 2005.

In *Proc. of the 11th Conf on Computer Generated Forces and Behavioral Representation*, May, 2002, Orlando FL.

4.1 Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion[1]

Benjamin Bell
Lockheed Martin Advanced Technology Laboratories
1 Federal Street, A&E Building, Camden, NJ 08102
(856) 338-4039
benjamin.l.bell@lmco.com

Eugene Santos Jr.
University of Connecticut, Dept of Computer Science & Engineering
UTEB, 191 Auditorium Rd., U-155, Storrs, CT 06269-3155
(860) 486-1458
Eugene@cse.uconn.edu

Scott M. Brown, Capt, USAF
Electronic Systems Center, Information Operations Program Office
240 Hall Blvd, Suite 105
San Antonio TX 78243-7058
(210) 977-3261
scott.brown@lackland.af.mil

ABSTRACT: Military and domestic security analysts and planners are facing threats whose asymmetric nature will sharply increase the challenges of establishing an adversary's intent. This complex environment will severely limit the capabilities of the classic doctrinal approach to diagnose adversary activity. Instead, a more dynamic approach is required - adversary decision modeling (ADM) - that, while a critical capability, poses a range of daunting technological challenges. We are developing methodologies and tools that represent a tractable approach to ADM using intelligent software-based analysis of adversarial intent. In this paper we present work being performed by our team (University of Connecticut, Lockheed Martin Advanced Technology Laboratories, and the Air Force Research Laboratory Human Effectiveness Directorate) toward a preliminary composite theory of adversary intent and its descriptive models, to provide a coherent conceptual foundation for addressing adversary decision processes, tasks, and functions. We then introduce notional computational models that, given own system-of-systems actions (movements and activities) and observations of an adversary's actions and reactions, automatically generate hypotheses about the adversary's intent. We present a preliminary software architecture that implements the model with: (1) intelligent mobile agents to rapidly and autonomously collect information, (2) information fusion technologies to generate higher-level evidence, and (3) our Intent Inference engine that models interests, preferences, and context.

[1] This work is supported in part by the Air Force Research Laboratory, Information Institute Research Project, Grant No. F30602-01-1-0595

1. Need for Adversary Decision Modeling

The United States military faces a world in which it will overmatch in size and armament most threats it may encounter. In order to make up for these mismatches, the asymmetric threat will be creative in its approach to warfare. These threats may be distributed, performing small strategic attacks across specific points within a wide area. The short timeframes and uncertain intentions of adversaries representative of 21st century operations will force planners to forego a predetermined "menu" of targets and instead dynamically select targets from among a more general list based on current priorities and opportunities. This mode of planning is inimical to target-based approaches, and it strains the objectives-based approach (at least as that approach has been employed to date). Within this complex environment, the use of static doctrinal reasoning in ascertaining adversary actions and reactions offers severely limited utility. Instead, a more dynamic approach to inferring and tracking adversary intent is needed.

The basis for such an approach is emerging from USAF-sponsored research. Dr. Maris "Buster" McCrabb has recently submitted a draft CONOPS [1] for a novel approach to planning, executing, and assessing military operations. This approach — termed effects-based operations (EBO) — is the best candidate to serve as the basis of the operations model we require. EBO is a "…set of processes, supported by tools and done by people in organizational settings, that focuses on planning, executing and assessing military activities for the effects they produce rather than the targets or even objectives they deal with." ([1], p. 3) As such, EBO is framed with respect to outcomes produced (and/or predicted to be produced) in the battlespace.

One of the greatest technological challenges for the EBO approach is that of *adversary decision modeling* [2]. An example of a composite adversary decision model based on Llinas et al. ([3], [4]) is given in Whitaker and Brown [5] as shown in Figure 1.

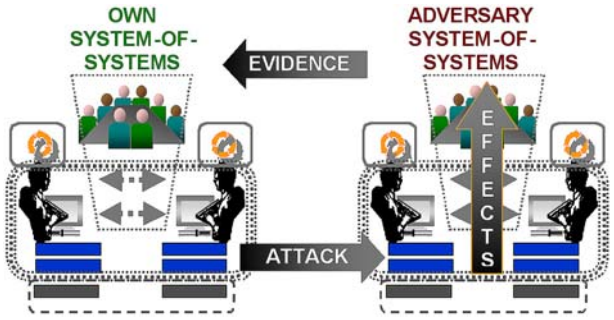


Figure 1. Composite adversarial decision model

We have defined five broad research issues raised by adversary decision modeling:

1. What critical human factors (e.g., interests, technology, emotions, knowledge and/or expertise, culture, environmental context) must be modeled?
2. What knowledge representations are necessary and sufficient for effective and efficient adversarial intent inference (i.e., how do we model the human factors and at what level of abstraction)?
3. How do we perform the necessary initial and subsequent knowledge acquisition (to include obtaining observational cues of effects) for prescriptive adversarial decision models given the non-cooperative nature of the adversary?
4. How do we use the models to inform friendly decision makers and increase their predictive battlespace awareness?
5. How do we test and evaluate our models to determine how well they perform?

In the remainder of this paper, we articulate the challenges facing adversary decision modeling and introduce an approach to overcoming these challenges that employs intent inference, probabilistic reasoning, information fusion and intelligent agents.

2. Background

Deriving hypotheses about future actions of an adversary requires *information* about the adversary's current actions and *inferences* about the adversary's motivations. The informational requirements can be approached by bringing data collection and fusion capabilities to bear; the inferential requirements can be approached by creating models to generate *descriptive* probabilities (to what extent does motivation X account for the set of observations Y?) and *predictive* probabilities

(how likely is future action Z given motivation X?). Before discussing the details of our approach, we provide a brief synopsis of these informational and inferential capabilities.

2.1. Informational Elements

Inference depends centrally on information, or evidence, and the more complete and accurate the information, the more reliable and predictive the inference. We employ a two-step process to generating a critical mass of useful information. In Step 1, intelligent agents are dispatched to autonomously migrate across own-force networks and collect information relevant to a specific need. An agent can then collect whatever information it encounters on the spot, or post itself as a "sentinel," persistently monitoring a data source until its criteria are met for migrating back with the collected information. The second step is to combine (or "fuse") the collected information, to merge information from different sources that describes the same event or entity, and to reconcile contradictory information.

2.1.1. Intelligent Agents

Agents are autonomous software processes that can migrate throughout a network, execute instructions, make decisions, collect data, and return to the original host. Intelligent agents can play a key role in adversary decision modeling by providing a stream of observed events to the intent inference mechanism. Lockheed Martin Advanced Technology Laboratories (LM-ATL) and the University of Connecticut (UConn) have employed agents for a variety of purposes, among them: to collect and organize command and control information ([6], [7]); to gather and monitor information in intelligence databases [8]; to perform multiple mission planning and execution activities [9]; and to assist military personnel in placing and tracking supply requests [10]. Intelligent Agents play a key role in our approach to adversary decision modeling, actively tracking down relevant information to provide the raw data needed to form conclusions about an adversary's plans.

2.1.2. Information Fusion

Information gathered by intelligent agents may be derived from heterogenous sources, have multiple modalities, be contradictory and incomplete, and span a temporal range. Sophisticated information fusion capabilities are required in order to transform what the agents gather from a raw form to an integrated, consistent and complete form. Information fusion can occur at multiple levels of abstraction. A widely adopted lexicon advanced by the Joint Directors of Laboratories has characterized four levels of fusion as object refinement (Level 1), situation refinement (Level 2), threat refinement (Level 3), and process refinement (Level 4).

Work performed by LM-ATL under the Army's Rotorcraft Pilot's Associate program has resulted in a technology base for Level 1 and Level 2 fusion. This technology base provides a fusion architecture and near real-time fusion engine for handling multi-sensor, multi-track fusion ([11], [12]).

On-going efforts in our group are exploring the integration of fusion and agents for situation awareness applications ([13], [14]). Our approach to adversary decision modeling seeks to leverage this promising synergy.

2.2 Inferential Elements

Creating a fused picture that reflects the current tactical situation provides the critical inputs from which to reason about an adversary's intent. Our approach to this second phase relies on *user-intent inference*, to generate expectations about goals based on observations, and *adversary intent inference*, to propagate those expectations through a reasoning engine that considers not only observed actions and their corresponding goals but inferred motives as well.

2.2.1. User Intent Inference

User Intent Inference (hereafter called simply "intent inference") involves deducing an individual’s goals based on observations of that individual’s actions [15]. In automated intent inference, data representing observations of an individual, the individual’s actions, or the individual’s environment (collectively called *observables*) are gathered by direct and indirect mechanisms. These data are then fused to merge the raw information into higher-level constructs. Finally, the fused information is interpreted against one or more behavioral models, constructed and optimized for the individual’s behavior patterns. The models match the observed and fused

information against patterns of behavior and derive inferred intent from those patterns.

Intent Inference can be employed to provide three kinds of hypotheses [16]. *Descriptive* intent inference provides insight into the motivations behind actions that have just occurred. *Predictive* intent inference can anticipate future actions given the individual's inferred goals. *Diagnostic* intent inference detects deviations between predicted and observed actions to reveal possible user errors.

User intent inference involves the use of observations of an individual's use of an application or system to automatically infer the user's goals with respect to the user's tasks. These data can be direct observations of human-system interface interactions or indirect observations of the user's environment, including operator biometrics, interactions between operators, and inputs from decision support tools. The results of the inference can be used to proactively aid the user, balance the user's workload, filter incoming information, or monitor the user's task progress.

UConn has continued to refine the work on the Core Interface Agent (CIA) Architecture [17]. CIA uses Bayesian networks to infer intent given sets of observables. CIA has been used to infer user intent in an expert system shell called PESKI ([18], [19], [20]) and in an intelligent information retrieval system called Kavanah ([21], [22]). In the former application—the probabilistic expert system development environment (PESKI)—a suite of intelligent knowledge engineering tools (agents) was developed to help the knowledge engineer construct intelligent systems that managed uncertainty. The ultimate goal for PESKI is to guarantee that all actions taken by the expert and machine in building a decision support system are done as efficiently as possible, always consistent, and always correct. Given that knowledge engineering is rife with many incremental choices and alternatives at each stage, making the right choices by the human/machine is paramount. Employing user intent, an intelligent user interface was developed for PESKI that attempted to help select the appropriate tools for the expert. This project led to the development of CIA that demonstrated that the ability of user intent inference to adapt to changing situations and contexts is a critical component even in very controlled/targeted domains such as building specialized decision support systems.

The main goal of Kavanah is to use the interface agent (also called active user interface) to assist the users in getting the right information at the right time using the right tools. The principles behind the design of the system are efficient construction of a model of the user's long and short-term interests and preferences, dynamic reasoning from the user's information seeking context, and applying decision theoretic principles and probabilistic reasoning techniques wherever they are appropriate. We clearly separate the concepts of interests and preferences in a dynamic fashion. The term *interests* denotes the topics or subjects that the user is focusing on in the information-seeking task. The term *preferences* denotes how the user would go about acquiring and viewing the desired information. The interface agent in Kavanah proactively constructs the queries on the user's behalf as it learns the user's style in searching. It also updates the knowledge base to incorporate the new knowledge that it has learned from the user's interactions with the system.

Recently, LM-ATL adapted the CIA Architecture for use in team intent inference. Team intent inference is used to deduce the goals of a group based upon observations of the individuals within that group. Although individual intent is used to infer team intent, this team intent can also be used to help infer the intent of other individuals within the group, forming a natural mutual reinforcement positive feedback loop. Team intent inference can also be valuable in diagnosing coordination lapses within a group, such as two members of the group working at cross-purposes or multiple members duplicating actions. In addition, team intent inference, when combined with rigorous individual descriptive intent inference, can be used to identify oversights in a team's plans. This new architecture has been tested within a theater ballistic missile defense (TBMD) domain in a time-critical targeting demonstration application [23]. The demonstration prototype, named Observer, performs both descriptive and predictive intent inference for a team of operators within a TBMD cell. Observer monitors task progress in the identification of TEL (Transporter-Erector-Launcher) activity and aids cell analysts in the assignment of strike assets against TELs by tasking mobile intelligent agents with proactive information queries within context. The use of team intent inference provides fast coordination between operators in different parts of the cell.

2.2.2. Adversary Intent Inference

User Intent Inference can help answer *what* the individual is doing, *how* the individual might do it, and *why* the individual is doing it. If we substitute the thinking of an adversary for that of a user, we can consider models that might tell us what an adversary's actions suggest he might do in the future. In other words, we can adapt the *User* Intent Inference approach to something we can label *Adversary* Intent Inference. We can thus leverage the natural isomorphism between our prior work in the field of user and team intent inference and the domain of adversary intent inference. While the operational world surrounding an intent inference application would be very different, the inner mechanisms of intent inference map directly between domains. Note that this does not claim to solve the complete unified adversary intent problem. Instead, we offer our approach as a conceptual foundation on which to build a larger unified theoretical solution in the future.

While adversary intent could be construed as having as its overall goal the ability model the enemy in its entirety (stated as “enemy-as-a-system” in the EBO CONOPs), we believe that a necessary starting point is to model an *adversary commander’s intent*. This is a much more manageable and feasible task based on currently available technology and research developments especially in intent inference. Once enemy commander intent is suitably modeled and captured, we can then compose these individual intent models into larger collectives using our work in team intent modeling to address the general problem of adversary intent inference (“enemy-as-a-system").

The next section highlights some of the challenges that adversary intent inference poses and describes our approach to overcoming those challenges.

3. User Intent to Adversarial Intent

User Intent Inference is the basis for Adversary Intent Inference, so we can use the components of the former as analogs to help structure the latter. Recall from our discussion of User Intent Inference that the process relies on: (1) mechanisms for capturing observations of the user's environment, and interactions between the user and the automation systems; (2) a process for interpreting raw information and generating higher-level (fused) constructs; and (3) one or more behavioral models of the user.

Analogizing to Adversary Intent Inference, we require mechanisms to capture events in the environment, algorithms to fuse raw event reports into a common picture, and one or more behavioral models of the adversary. Although User Intent Inference is itself a difficult enterprise, we have in the adversarial instance greater complexities, reduced access to information, and the likelihood of stealth and deception.

In discussing these challenges, it is important to point out that the intelligence community already performs a kind of adversary intent inference, relying heavily on the skills and intuitions of experienced analysts. In fact the steps defined in the intent inference process have corollaries to the intelligence cycle: capturing observables is referred to in the intelligence community as "collection", fusing the information is closely aligned with "processing", and matching those data against behavioral models is the process of "analysis and production."

3.1. Collection Challenges

Consider first the "observables" that must be captured and interpreted. While observables in the user intent domain stem from data collected from human use of systems, observables in the adversary intent domain take the form of tactical information derived from intelligence databases, observations of the tactical environment, and input from online human experts. In place of window events, keystrokes, and mouse movements, our system in the adversary intent domain will use information about adversary location, movements, and activities to drive its inference.

An inherent problem associated with directly observing an adversary's actions is that an adversary probably does not want his actions observed; a corollary is that actions that cannot be hidden from observation may be masked by conducting similar or co-occurring decoy operations. In addition to the adversary's wishes to confound direct observation, it is often the case that the actions themselves are taking place in areas beyond the reach of direct observation, or in environments where direct observation is difficult. These challenges (addressed by the intelligence community's collection

organizations) add a layer of complexity to the adversary intent inference.

3.2. Processing Challenges

Next consider the process of combining raw data into higher-level information. Merging user intent data might consider the state of the application, analyses of information queries, and the content of user dialogue with team members. In the adversarial domain, our system must integrate facts about the local terrain, regional weather, and the salient political climate.

Intelligence analysts who perform this task will examine raw information from multiple sources and ask questions like "are these multiple indicators reacting to the same event or multiple events?", "what does indicator A mean in the context of indicator B?", and "what intermediate conclusion can be drawn from the co-occurrence of these multiple data?" Analysts rely on access to information collected by other intelligence assets and on their own knowledge, expertise, and intuitions to properly process the raw information.

Because information fusion has evolved as an intelligence capability, adversary intent inference does not introduce entirely new challenges. The range of information to be combined, however, is likely to be quite broad, across multiple domains, sources, and levels of abstraction. Merging information from battlefield sources, national assets, and news feeds publicly distributed on the web, for instance, is likely to present some combinatorial problems.

3.3. Analysis and Production Challenges

Developing predictive models of adversary behavior requires detailed histories, comprehensive digests of the intelligence picture, and highly experienced analysts. The intelligence community devotes substantial resources to developing predictions of adversary behavior (principally as a "gray matter" rather than automated process). This "analysis and production" stage is made far more difficult by asymmetric threats, as adversaries need no longer be sovereign states and can exhibit a breadth of political goals that extend beyond conventional sources of international conflict such as border disputes or control of resources.

Finally consider behavioral models. For adversary decision modeling, tactical goals will replace the software application goals (in the user intent case) that result from the intent inference process. Descriptive intent inference in this case would result in identification of an adversary force's objectives and, given models of tactical reasoning, could recommend appropriate reactions. Predictive intent inference would indicate expected activity by the adversary and explain the reasons behind that activity. Diagnostic intent inference could produce alerts of attempted subterfuge or uncover missteps on the part of the adversary.

Similarly, intent inference of echelons of adversary forces provides advantages analogous to those arising from team intent inference. Identification of the goals of one adversary group can be used as a discriminator in identifying the goals of other subsets of the adversary. Intent inference across groups of the adversary could also result in the discovery of breakdowns among those groups – knowledge that can be used for tactical advantage.

4. Approach

Our discussion in the previous section discussed how user intent inference could form the basis for an

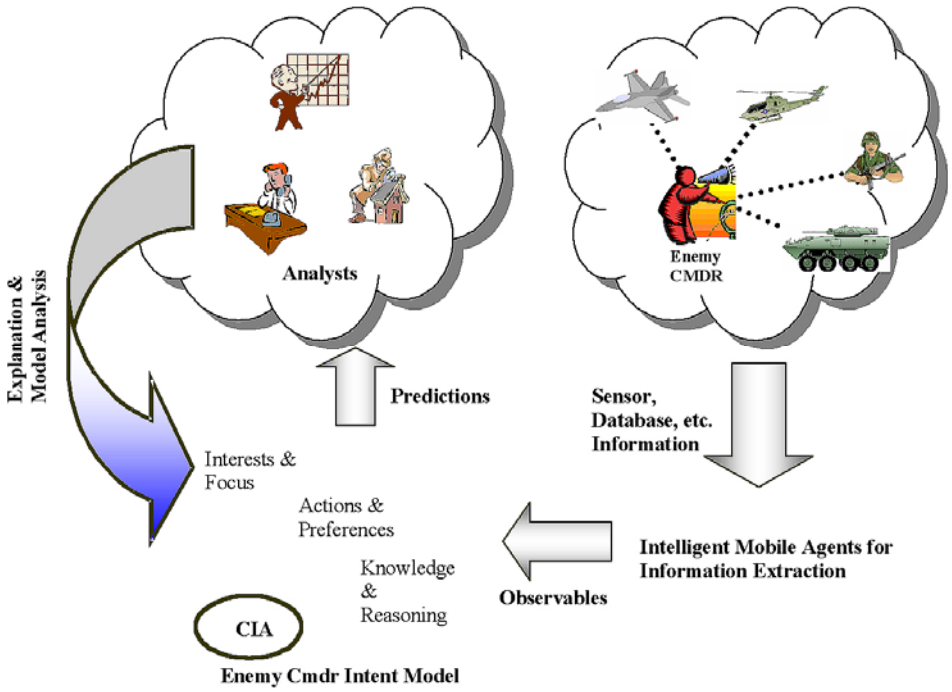


Figure 2. Adversary Intent Inference architecture

adversary intent approach, and outlines some of the challenges unique to the adversarial domain. In this section we summarize our overall approach (Figure 2) and the specific technologies we bring to bear on each of the three phases in the intelligence cycle: collection, processing, and analysis and production.

4.1. Intelligence Collection with Intelligent Agents

Automation support for collection is needed to extend the reach of collection devices and analysts without overextending the human capital required to monitor and gather the data. Instead, analysts require intelligent agents, who can be tasked to monitor one or more information sources and report back when some set of conditions is satisfied. In our approach, intelligent agents provide this persistent monitoring capability. Agents can be created dynamically (by a user or by another agent) and given instructions specifying where to go, what do to upon arrival at the remote host, and what (and when) to transmit information back. Deploying large numbers of intelligent agents addresses the acute collection problems presented by the adversarial domain.

4.2. Intelligence Processing with Information Fusion

Providing automation support for processing of raw information requires systems that can reason about the domain(s) under examination as a prerequisite to combining information drawn from that domain. At lower levels (Level 1 Fusion), data can be merged on the basis of algorithmic approaches with less emphasis on domain knowledge. As information to be combined moves up the abstraction hierarchy (Level 2 Fusion), reasoning plays a more prominent role. In our previous fusion work under the Rotorcraft Pilot's Associate program, for instance, combining track data required both a fusion algorithm and a domain ontology ([11], [12]).

Our approach to overcoming the combinatorial problems likely to be encountered in the adversarial domain rests on carefully controlling how information gets combined via a Threat Evidence Monitor (TEM). The TEM receives requests from the inference element (Bayesian Network; see next section) and then tasks intelligent agents with collecting information relevant to a given request. Information returned by the agents is fused by integrating our Level 1/Level 2 fusion capabilities,

domain ontologies, and semantic web technologies being developed for the DARPA Agent Markup Language (DAML) program. One objective of DAML is to provide for semantically-tagged web content, and to create a rich set of ontologies for defining the entities and relationships that are representing in DAML-formatted web pages and databases. The agents tasked by the TEM can then retrieve information from battlefield sources as well as from web pages. After applying its domain-guided fusion process, TEM sends the fused information back into the Bayesian Network.

4.3. Analysis and Production with Adversary Decision Modeling

To help provide automation support for the Intelligence Analysis and Product process, our approach employs user ontology networks (as represented by Bayesian Networks) as a tool for representing the possible decision making behavior of an adversary, while capturing and reasoning about the probabilistic nature of the process. Our model consists of the three components discussed earlier: Interests, Preferences, and Context. Using the observables and inputs from our human analysts, this model performs our intent inference and updates the enemy commander intent model as needed via the CIA sub-module.

We can also apply our team intent inference approaches to collections of enemy commander intent models. The hierarchical structure of team intent models maps naturally to the hierarchical nature of military command. By performing team intent inference with respect to echelons of enemy commanders, we can better understand the goals of individual commanders, identify particularly important objectives (when multiple commanders work toward the same small-scale goal), and detect breakdowns in the enemy's communications (when multiple commanders work toward cross-purposes).

5. Current Status & Future Work

We have outlined the need for adversary intent inference in order to effectively confront asymmetric threats and to support emerging doctrine such as Effect Based Operations. Our previous research in User Intent Inference was presented as a basis for Adversarial Intent Inference that applies similar techniques but which also poses unique and serious challenges. To better reveal where our technologies fit and where they need to be expanded, we summarized the difficulties of inferring an adversary's intent, broken down into the discrete phases of the intelligence cycle. We then introduced the specific technologies we bring to bear on each phase as part of our overall approach to adversary intent inference.

Our work is in the definition phase and this document reflects both our past experience and current plans. In the coming weeks and months we will engage with various end-user groups in the intelligence community to build a preliminary adversary intent model and to identify the information sources to be accessed. A preliminary prototype will be developed during the first year.

6. References

[1] McCrabb, M., Concept of Operations for Effects-Based Operations, Draft paper for AFRL/IFTB, Version 2.0 of June 2000

[2] McCrabb, M., and Caroli, J. "Behavioral Modeling and Wargaming for Effects-Based Operations," Workshop on Analyzing Effects-Based Operations, Jan, 2002, McLean, Virginia, Military Operations Research Society.

[1] Llinas, J., Drury, C., Bialas, W., and Chen, A.C., "Studies and Analyses of Vulnerabilities in Aided Adversarial Decision Making," AFRL-HE-WP-TR-1998-0099, 1998.

[3] Llinas, J., Drury, C., Jian, J. Y., Bisantz, A., and Younho Seong, Y., "Studies and Analyses of Aided Adversarial Decision Making Phase 2: Research on Human Trust in Automation," AFRL-HE-WP-TR-1999-0216, 1999.

[4] Whitaker, R. D., & Brown, S. M., Addressing Information Operations Research from a Cognitive Engineering Perspective, unpublished technical report, Air Force Research Laboratory, Wright-Patterson Air Force Base, Ohio, March 2001.

[2] Hofmann, M., Chacon, D., Mayer, G., and Whitebread, K. "CAST Agents: Network-Centric Fires Unleashed." 2001 National Fire Control Symposium: Session: Automation of the Kill Chain, August 27-30 2001, Lihue, Hawaii.

[3] McGrath, S., Chacon, D., and Whitebread, K. "Intelligent Mobile Agents in the Military Domain." In Proc. of the Fourth International Conference on Autonomous Agents 2000, June 2000.

[4] Whitebread, K. and Jameson, S., “Information Discovery in High-Volume, Frequently Changing Data,” IEEE Expert Journal – Intelligent Systems and Applications, May, 1995.
[5] Saba, M.G., and Santos Jr., E. “The Multi-Agent Distributed Goal Satisfaction System,” Proceedings of the International ICSC Symposium on Multi-Agents and Mobile Agents in Virtual Organizations and E-Commerce (MAMA 2000), 389-394, Wollongong, Australia, 2000.
[6] Daniels, J., and Bell, B. Listen-Communicate-Show (LCS): Spoken language command of agent-based remote information access. In Proc. of Human Language Technology Conference, HLT- 2001, San Diego, CA, Mar, 2001.
[7] Hofmann, M. "Multi-Sensor Track Classification in Rotorcraft Pilot's Associate Data Fusion," American Helicopter Society 53rd Annual Forum, May 1997.
[8] Malkoff, D., and Pawlowski, A. "RPA Data Fusion." In Proc. of the 9th National Symposium on Sensor Fusion, Infrared Information Analysis Center, September 1996.
[9] Jameson, S. "Architectures for Distributed Information Fusion To Support Situation Awareness on the Digital Battlefield." In Proc. of the 4th International Conference on Data Fusion, August 2001.
[10] Pawlowski, A., and Stoneking, C. "Army Aviation Fusion of Sensor-Pushed and Agent-Pulled Information." Presented at the American Helicopter Society 57th Annual Forum, Washington, DC May 9-11, 2001.
[11] Geddes, N., "The Use of Individual Differences in Inferring Human Operator Intentions," Proceedings of the Second Annual Aerospace Applications of Artificial Intelligence Conference, 1986.
[12] Franke, J., Brown, S.M., Bell, B., and Mendenhall, H., “Enhancing Teamwork Through Team-Level Intent Inference,” Proceedings of the International Conference on Artificial Intelligence, 2000.
[13] Brown, S.M., Santos, E., Jr., and Banks, S.B., “Utility Theory-Based User Models for Intelligent Interface Agents”, Proc. of the 12th Biennial Conf. of the Canadian Society for Computational Studies of Intelligence, 379-393, 1998.
[14] Santos, E., Jr. and Santos, E.S., “A Framework for Building Knowledge-Bases Under Uncertainty,” Journal of Experimental and Theoretical Artificial Intelligence 11, 265-286, 1999.
[15] Brown, S.M., Santos, E., Jr., and Banks, S.B., “Active User Interfaces for Building Decision-Theoretic Systems,” Proceedings of the 1st Asia-Pacific Conference on Intelligent Agent Technology, 244-253, Hong Kong, 1999.
[16] Brown, S. M. and Santos, E., Jr., “Active User Interfaces”, IDIS TR No. 101, Intelligent Distributed Information Systems Laboratory, University of Connecticut, 1999.
[17] Santos, E., Jr., Brown, S.M., Lejter, M., Ngai, G., Banks, S.B., and Stytz, M.R., “Dynamic User Model Construction with Bayesian Networks for Intelligent Information Queries,” Proceedings of the 12th International FLAIRS Conference, 3-7, 1999.
[18] Santos, E., Jr., Brown, S.M., and Nguyen, H., “Medical Document Information Retrieval Through Active User Interfaces,” Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI '2000), Las Vegas, NV, 2000.
[19] Bell, B., Franke, J., and Mendenhall, H., “Leveraging Task Models for Team Intent Inference,” Proceedings of the International Conference on Artificial Intelligence, 2000.

Author Biographies

BENJAMIN BELL is in the Artificial Intelligence Laboratory at Lockheed Martin Advanced Technology Laboratories, where he investigates intelligent interactive systems, decision support and training. He previously served on the faculty of Teachers College, Columbia University.

EUGENE SANTOS, JR. is currently Interim Director for The Taylor L. Booth Research Center at the University of Connecticut and an Associate Professor of Computer Science and Engineering in the Computer Science and Engineering Department. His areas of research interest include multi-agent systems, intelligent information systems, decision support, intent inference, human-computer interaction, intelligent user interfaces, uncertainty, and probabilistic reasoning.

SCOTT BROWN is currently the lead engineer for the Information Warfare Planning Capability (IWPC). He previously served as the program manager of an advanced research development program at the Human

Effectiveness Directorate, Air Force Research Laboratory. His research interests include knowledge elicitation and representation, intent inference, information visualization, and intelligent user interfaces.

4.2 Knowledge Acquisition for Adversary Course of Action Prediction Models

Scott M. Brown, Major, USAF

Electronic Systems Center, Information Operations Program Office
240 Hall Blvd, Suite 105
San Antonio TX 78243-7058
scott.brown@lackland.af.mil

Eugene Santos Jr.

University of Connecticut, Dept of Computer Science & Engineering
UTEB, 191 Auditorium Rd., U-155, Storrs, CT 06269-3155
eugene@cse.uconn.edu

Benjamin Bell[1]

CHI Systems, Inc.
716 N. Bethlehem Pike, Suite 300
Lower Gwynedd, PA 19002-2650
bbell@chisystems.com

Abstract

Intelligence Preparation of the Battlespace (IPB) is a predominantly "gray matter-based" fusion and information synthesis process conducted to predict possible future adversary courses of action. The purpose is to understand where the enemy is in the battlespace, and to infer what we believe he will do next. From that understanding, military commanders plan their own course of action. As the state of the art improves, we are in a position to begin applying technologies to move the labor-intensive parts of IPB to the computer, allowing the planner to perform those tasks that are more suited to human capabilities. This is a primary focus of our research effort. This paper presents the approaches that we are adopting to acquire the knowledge necessary to build models to assist decision makers determine adversary intent. We discuss how the IPB process can assist with knowledge acquisition and we present a detailed discussion of our AII system and the techniques we have developed to collect and process the data necessary to map observations of the adversary into evidence to support reasoning about the adversary's intent.

Introduction

"The **human decision-making processes** are the ultimate target [sic] for offensive [information operations]." (Joint Publication 3-13, 1998, original emphasis). To effectively target these decision-making processes involves a focused planning, execution and assessment process that adroitly acts in anticipation of a threat's courses of action (COAs) by keying in on the relevant aspects of the battlespace. This *Predictive Battlespace Awareness* "…involves studying an adversary to understand what he'll do, how he'll do it, what his capacity to inflict harm will be, and the environment in which he is operating — in short, knowing the scene of the crime before

the crime is committed." (Behler 2001). Determining an adversary's[2] course of action involves identifying, evaluating and prioritizing the adversary's likely objectives and desired end state, and the full set of COAs available to the threat.

An emerging technology for predicting adversary courses of action is adversarial decision modeling (ADM). ADM focuses on modeling key adversary decision processes, objectives, centers of gravity, and high value targets from a number of perspectives including socio-cultural, political and economic. Adversarial decision modeling raises several broad research issues:

1. What critical human factors (e.g., situational awareness, interests, risk propensity, emotions, knowledge and/or expertise, culture, environmental context) must be modeled?
2. What knowledge representations are necessary and sufficient for effective and efficient adversarial decision modeling (i.e., how do we model the human factors, at what level of abstraction do we model these factors and how do human factors influence adversary courses of action?)?
3. How do we perform the necessary initial and subsequent knowledge acquisition (to include using subject matter experts) to build adversarial decision models, given the non-cooperative nature of the adversary?
4. How do we use these models to support friendly decision makers and to increase their predictive battlespace awareness?
5. How do we test and evaluate our models to determine how well they perform in a broad range of situations?

We are addressing the issues listed above through an Information Institute Research Project (IIRP) funded by the Air Force Research Laboratory's (AFRL) Information Directorate and performed in collaboration with AFRL's Human Effectiveness Directorate. Specifically, we are investigating those salient human factors characteristics that must be modeled (issue #1); the creation of efficient and effective computational models (issue #2) that, given observations of an adversary's actions and reactions (issue #3), generates hypotheses about the adversary's intent and suggests appropriate responses (issue #4). The efficacy of our models within cooperative domains has already been proven (Bell, Franke, and Mendenhall, 2000; Franke, et al. 2000). The value-added of our models to Department of Defense (DoD) personnel performing adversary course of action prediction will be iteratively evaluated during the IIRP (issue #5).

Adversarial decision modeling technology yields ideas about what the adversary is trying to accomplish, as well as explanations about why the adversary is trying to accomplish those particular objectives. Deriving hypotheses about future actions of an adversary requires *information* about the adversary's current actions and *inferences* about the adversary's motivations. The informational requirements can be approached by bringing together knowledge elicitation, data collection and data fusion capabilities. The inferential requirements can be approached by creating models to both generate *descriptive* probabilities (to what extent does motivation X account for the set of observations Y?) and *predictive* probabilities (how likely is future action Z given motivation X?). The results of this system can be communicated to situation assessment tools to further refine the overall operational picture within a particular tactical setting. Further discussions of the adversary models and information collection and fusion issues can be found in Bell, Santos Jr. and Brown (2002).

Background

[2] Throughout this paper we use the term "adversary" broadly to mean non-friendly forces that include the actual enemy, non-participants, etc.

Intelligence Preparation of the Battlespace and COA Development

A key process used by the United States military to predict adversary courses of action is the Intelligence Preparation of the Battlespace (IPB) process. The goal of this doctrinally driven, four-step process is to "…reduce uncertainties concerning the enemy, environment, and terrain for all types of operations." (Joint Publication 2-01.3 2000). This is achieved by determining the adversary's likely COA, describing the environment friendly forces are operating within and the effects of this environment on these forces ability to achieve their goals. The four-step process is briefly provided below:

1. **Define the battlespace:** assess the crisis situation; review commander's guidance / objectives; identify limits of operational area, area of interest, significant battlespace characteristics; evaluate existing databases and identify information gaps**;** obtain products / information required to conduct remainder of IPB
2. **Describe battlespace effects:** describe how characteristics of the surface, aerospace, information, human and weather dimensions affect operations employment; identify information gaps
3. **Evaluate the adversary:** map relevant adversary processes and identify friendly and adversary centers of gravity, capabilities, limitations and vulnerabilities; perform critical nodes analysis to identify high value targets; identify information gaps
4. **Determine adversary COAs:** identify, evaluate and prioritize the adversary's likely objectives and desired end state and the full set of COAs available to the adversary while identifying initial information collection requirements

The IPB process, as part of operational environment research, is a major input into (both friendly and adversary) objective determination for deliberate and crisis action planning, and to a lesser extent, force execution due to the time and information demands of the IPB process.

COA development[3] begins with reviewing Combatant Commander guidance, intent and objectives and then constructing a strategy-to-task framework. This framework allows planners to determine supporting objectives (e.g., service component objectives that support the commander's objectives), tasks, actions, and targets based on the IPB information they have previously generated on a particular adversary.

[3] There are two COA development activities being performed simultaneously—one to determine how our time-phased actions will meet a commander's objectives and one to determine what actions an adversary might take to achieve his objectives. Operational planners typically do the former while intelligence analysts do the latter. To avoid confusion, we will always fully designate the adversary COA development process; otherwise, we will mean the friendly COA development process. To a large extent, the products generated are the same.

For each level in the strategy-to-task decomposition hierarchy, success indicators (sometimes called battle damage indicators or measures of success (MOS)[4]) are assigned as observables with quantitative and / or qualitative metrics. During this COA development-planning phase, the IPB information is refined and additional information is collected.

Objective: Gain & maintain air superiority forward through the main battle area (MBA)
MOS: Enemy sorties negligible.
Task: Degrade IADS C2 and air surveillance capability
MOE: AOCs reduced to RF communications within MBA
MOE: Air defense weapon systems forced into autonomous operations throughout the MBA
Action: Destroy key IADS nodes
MOP: IMINT confirmation

Objective: Isolate leadership and prevent effective directive of MBA air defense forces
Task: Disrupt the ability of key military leaders to orchestrate conflict
MOE: Leaders isolated from ISR and space assets
MOE: Leadership unable to use land communications
Action: Destroy key C2 links and nodes supporting MBA forces
MOP: IMINT and SIGINT confirmation

Task: Mislead MBA decision makers
MOE: False posturing of MBA forces
MOE: SIGINT confirmation
Action: Feign major operations south of MBA
MOP: Troop movement out of MBA

Figure 1. Strategy-to-Task Hierarchy Example

An example of a fictitious strategy-to-task hierarchical decomposition is provided in Figure 1 below. Numerous tools support this strategy-to-task decomposition including AFRL's Effects-Based Operations Advanced Technology Demonstration Strategy Development Tools, Joint Information Operations Center's Information Operations Navigator, and Electronic Systems Center's Information Warfare Planning Capability.

It should be noted, however, that few tools exist that explicitly capture the adversary's COAs. A notable exception is the Target Prioritization Tool for Links and Nodes (TPT-LN). The use of a strategy-to-task hierarchy to represent adversary COAs has merit for military planners. The advantages include re-using a well-known process and representation (i.e., a hierarchical decomposition) and the ability to explicitly compare and contrast friendly force COAs with competing adversary COAs.

Effects-Based Operations and Adversary Intent Inferencing

[4] Measures of success attach to a desired end state (that is, what the situation should look like once the operation is over). Since the objective is typically accomplished over time, the end state is decomposed into a series of phased events. An event should have a MOS. Sets of conditions determine whether the event has occurred. Measures of effectiveness and performance (MOEs and MOPs) attach to conditions.

Target-based and objectives-based (the well-known "strategy-to-task" approach described above) approaches to planning do not explicitly address the adversary's decision-making processes. A new approach to planning that explicitly addresses the adversary must be developed. The basis for such an approach is emerging from USAF-sponsored research. This approach, termed effects-based operations (EBO), is the best candidate to serve as the basis of the operations model we require (McCrabb 2000). Basically stated, EBO is "an approach that…explicitly seeks to understand, trace, and anticipate direct and indirect effects of a specific action…on an adversary's course of action." (Fayette 2001) EBO is framed with respect to outcomes produced (and / or predicted to be produced) in the battlespace. *EBO inherently addresses an adversary as a system*. The notion of "effect" is predicated upon the presumption that there is an object of reference (specifically one systemically organized), namely the adversary, whose state(s) can be identified and influenced through prospective courses of action. EBO planning is predicated on a coherent model of the state(s) and dynamics of the adversary system(s). At the center of the EBO concept is the idea that effective friendly COA planning can and should be framed with respect to effects to be induced in an adversary system.

The key to effects-based operations revolves around determining how an adversary *should / can / could* react to system perturbations resulting from actions on the battlefield from our own forces (McCrabb 2000). One of the greatest technological challenges for the EBO approach is that of adversarial decision modeling. While EBO's overall goal is to model the enemy in its entirety (stated as "enemy-as-a-system" in the EBO CONOPs and including the physical, data, cognitive, and social aspects of the battlespace centers of gravity and the dependency linkages between them), we believe that a necessary starting point is to model an *adversary commander's intent*. Intent[5] inference involves deducing an individual's goals based on observations of that individual's actions (Geddes 1986). In automated intent inference, this process is typically implemented through one or more behavioral models that have been constructed and optimized for the individual's behavior patterns. In an automated intent inference system, data representing observations of an individual, the individual's actions, or the individual's environment (collectively called *observables*) are collected and delivered to the model(s), which match the observables against patterns of behavior and derive inferred intent from those patterns. These inferences can then be passed to an application for generation of advice, definition of future information requirements, proactive aiding, or a host of other benefits. Furthermore, the success of adversary intent inferencing addresses a key technological barrier of EBO—that of the human element's impact in EBO. Once adversary intent is suitably modeled and captured, we can then compose these individual adversary commander's intent models into larger collectives using our work in team intent modeling (Franke, et al. 2000) to address the general problem of the "enemy-as-a-system".

A Knowledge Acquisition Approach

In this section, we detail our approach for performing the initial and subsequent knowledge acquisition to build adversarial decision models. Our approach is pragmatic in its use of pre-existing processes, tools, and data already in wide use by the DoD planning community. The construction of models to support adversary intent inferencing will be driven by a number of

[5] What exactly constitutes intent has long been debated in the cognitive and psychological sciences (as well as artificial intelligence). For purposes of this discussion we stand on the following military-oriented definition: Intent is composed of a commander's desired end-state/goal, the purpose/reason for pursuing that end-state, a methods/means to achieve the end-state, and a level of commitment to achieving that end-state (based on acceptable risk of the pursuit and probability of success).

different sources. The single most influential source of adversary intent modeling information must be the human subject matter experts (SMEs) who are most familiar with particular adversaries. Doctrinal knowledge can provide a foundation on which to build, but does not offer a complete solution. These SMEs can provide our models with the intuitive reasoning that cold facts and rigid doctrine cannot. In addition, military planners must also address historical case studies of the adversary and up-to-date information of the political environment of the region(s) in question. By providing intuitive means for an SME to specify possible adversary objectives, the relationships between these objectives and the tasks and actions that constitute an adversary's courses of action, an adversary intent inferencing system can provide assistance with collecting and organizing command and control information (Hofmann et al., 2000; McGrath, Chacon, and Whitebread, 2000), gathering and monitoring information in intelligence databases (Whitebread and Jameson 1995) and performing multiple mission planning and execution activities (Saba and Santos Jr. 2000). One advantage to our pragmatic approach is that it allows for an incremental, phased approach to adversary course of action prediction. We fully realize that a model is only as good as the data that supports that model. As any particular situations "flares up" and military planners start the IPB process for an adversary, developing related intelligence for the area of interest and therefore learning as they go along, given there is little to no existing information on a given adversary.

Mapping the User's Domain to Adversary Intent Inferencing

While observables in the user intent domain stem from data collected from use of computer systems by humans, observables in the adversary intent domain take the form of tactical information derived from intelligence databases, observations of the tactical environment, and input from human experts interacting with the adversary intent models. In place of window events, keystrokes, and mouse movements common in the user intent domain, our system in the adversary intent domain uses information about adversary location, movements, and activities to drive its intent inference processes. In place of computer state, analyses of information queries, and the content of user dialogue with team members, our system bases inferences on facts about the local terrain, regional weather, and the salient political climate.

The modeling process begins by analyzing military planners who are performing the Intelligence Preparation of the Battlespace (IPB) process. These planners define the range of objectives (i.e., end-states or goals) that an adversary might attempt to carry out and the available actions (i.e., means / methods) that the adversary has for carrying out those objectives. These objectives and actions represent the space of possibilities that the intent inference system must explore in examining adversary behavior. The modeler must also identify the observables associated with each individual action and indicate the method by which each observable can be ascertained. This will guide the integration of the intent inference mechanism into an operational context.

Next, the modeler must choose appropriate system architectures to capture the relationships between objectives, actions, and the observables. Given the simple fact that intent inference is an inherently uncertain process, this system architecture must both be able to deal with uncertain (and incomplete) information, as well as adapt over time as the result of new (possibly previously unknown or unanticipated) information. Finally, the modeler must decide upon the expected use of the system and implement a surrounding framework to make use of the models' inferences. Should the system perform descriptive, predictive, or diagnostic functions? How will the intent inference system provide timely, beneficial assistance to a decision maker?

We address several aspects of the system architecture design, including knowledge acquisition using the IPB process, the adversary intent inferencing model and the collection and production of observables, in the following sub-sections.

Knowledge Acquisition Using The IPB Process. The Intelligence Preparation of the Battlespace (IPB) process, first presented in the Background section above, is used by military intelligence analysts to arrive at an estimation of the adversary's possible courses of action (COAs). The IPB process provides an excellent basis for our knowledge acquisition. The output of the last step of the IPB process is very similar to the output from our adversary course of action prediction system, namely an estimation of the intent of the adversary and how the adversary is likely to act into the future. By reviewing the current IPB process, conducted primarily by human analysts with only limited automation support, we believe that it will be possible to determine much relevant information for creating a functional adversarial decision modeling (ADM) system.

First, it is possible to enumerate the various data sources that military intelligence analysts typically access in order to perform the IPB process. The structured data sources have schema that can be reviewed to create lists of data fields, and the unstructured data sources can be noted as possible candidates for automated evidence extraction and link discovery. In addition, analysts are likely to have some common tactical picture display in front of them, driven by data fusion processes. Such data fusion outputs would also be provided to an ADM system. Analysis of the IPB process allows us to enumerate the inputs most likely needed to support automated ADM.

Second, IPB analysts combine the evidence provided to them in certain ways in order to arrive at a hypothesis of the adversary's intent. Operators begin this process by analyzing the adversary from a number of different perspectives. Perspectives include political, cultural (research being performed into so-called "cultural lens" lends credence to the idea that culture should be considered when modeling adversaries), personality, emotional, economic, technological, will-to-win, risk perception, fatigue and morale. For example, if the adversary forces have not received supply within a certain period of time, and have been almost continuously under attack, it is likely that their morale will be low. Likewise if enemy forces are only weakly allied with their leaders, they may not share the same will-to-win. IPB analysts perform a series of such reasoning steps within a prescribed set of perspectives. By enumerating these perspectives, we believe that it will be possible to ensure that an ADM system, employing and reasoning about many of these same perspectives, will perform analysis of adversary intent in a comprehensive and exhaustive fashion. The Air Force Research Laboratories has been researching a number of descriptive decision models for adversarial decision-making (See, and Kuperman 1997; Llinas, Drury, Bialas, and Chen 1998; Llinas, Drury, Jian, Bisantz, and Younho Seong 1999).

Third, once IPB analysts have collected data, combined it into evidence under one of the several analysis perspectives, they infer adversary intent and determine plausible adversary courses of action. In order to do this, analysts, through training and experience, develop rules to map observables of adversary actions and general background data (such as fact books on the adversary's country and culture) to categories of intent. At the broadest level such categories of actions may be advance, attack, retreat or defend. With further refinement, it is possible to say, for example, that destroying a particular bridge is the intent of the adversary under the broader category of attack. These rules, mapping inputs to predictions of adversary intent are the most valuable aspect of the IPB process to capture and distill for use in ADM. We recognize that although some of these rules are made explicit in the IPB process, many are refined over the course of time by human operators and will thus be harder to capture and incorporate in the ADM system.

Adversary Intent Inferencing Model. The components of our adversary intent inferencing model, and the interactions between these components, are shown in Figure 2 below. The three core components that comprise our architecture and functions are as follows:

- Goals: Prioritized short- and long-term goals list, representing adversary intents, objectives or foci
- Rationale: A probabilistic network, representing the influences of the adversary's beliefs, both about themselves and about us, on their goals and on certain high level actions associated with those goals
- Actions: A probabilistic network, representing the detailed relationships between adversary goals and the actions they are likely to perform to realize those goals

The goal component captures *what* the adversary is doing, the action component captures *how* the adversary might do it, and the rationale component infers *why* the individual is doing it. Due to the inherent uncertainty involved in adversary course of action prediction, we use Bayesian networks (Pearl 1988) as the main knowledge representation for the rationale and action networks. Each random variable (RV) involved in the Bayesian networks is classified into one of four classes: axioms, beliefs, goals and actions. Each RV class is described below:

(a) Adversary axioms–represents the underlying beliefs of the adversary about themselves (vs. beliefs about our forces). This can range from an adversary's beliefs about his or her own capabilities to modeling a fanatic's belief of invulnerability. Axioms typically serve as inputs or explanations to the other RVs such as adversary goals

(b) Adversary beliefs–represents the adversary's beliefs regarding our forces (e.g., an adversary may believe that the United States is on a crusade against them or that the United States is not carpet-bombing territory)

(c) Adversary goals–represents the goals or desired end-states of the adversary. These goals are defined as either short-term or long-term in a goals list. Further we partition goals into two types: abstract and concrete. Abstract goals are those that cannot be executed (e.g., preserving launchers, damage US world opinion, defeating US foreign policy). They are satisfied by other abstract goals and also by concrete goals. Concrete goals are executable goals (e.g., repositioning launchers, contacting ambassadors, storing military equipment in civilian structures). Concrete goals can only be satisfied by concrete goals

(d) Adversary actions – represents the actions of the adversary that can typically be observed by friendly forces.

Figure 2 also shows feedback and explanation paths within the adversary intent inference (AII) model. Feedback from a human analyst, although unlikely to be totally certain, can be extremely valuable to the AII model, correcting and extending its intent inferencing logic. Explanation capabilities are essential in order for intelligence analysts, using AII, to understand why the AII model has reached particular inferences. The analysts must be able to inspect the reasoning paths used by AII so that they can develop a level of confidence in the output of the AII model.

Collection and Production of Observables. There are two primary inputs to the adversary intent inferencing (AII) model—intelligence, surveillance, reconnaissance (ISR) data (observations of the actions of the adversary, collectively the *observables*) and direct analyst input. Unlike the majority of observables used to infer user intent, observables in the adversary intent domain cannot be pulled directly from an application. Instead, they must be discovered within the flow of information available to the decision makers at the strategic, operational and tactical levels of operations. An example of an observable might be that "the adversary is repositioning its SCUD launchers" or "an enemy tank unit is approaching one of the blue force's key logistics and supply bases". A large portion of these observables can be found in existing intelligence databases. Other observables may come from situation reports from fielded units or from online fact-sheets for the

region of interest or from news reports. Observables may be gathered through friendly force sensor systems. We use the phrase "sensor system" here to mean any system capable of collecting data on the adversary. Friendly force human intelligence operatives may also be considered to be sensor systems.

In collaboration with our ongoing research, our research partner, Lockheed Martin Advanced Technology Laboratories (ATL), is developing, under internal research and development funding, agent-based technology to collect sensor reports from three broad categories—tactical battlefield sensors, structured intelligence databases, unstructured data—process these reports and combine them to produce evidence to support higher level reasoning, and in particular to provide inputs to the adversary intent inferencing (AII) model. This technology is referred to as the Smart Agent Generation Engine or SAGE and is based on ATL's Extendable Mobile Agent Architecture (EMAA) developed over the last seven years with DARPA and internal ATL funding (Whitebread and Jameson 1995; Hofmann et al. 2000; and McGrath, Chacon, and Whitebread 2000).

SAGE begins by analyzing the current evidence requirements of the AII model. By reviewing the state of the AII, SAGE attempts to prioritize evidence collection based on an analysis of which evidence will most aid with the disambiguation of the adversary's intent. Once SAGE has this prioritized list of evidence requirements, it begins to decompose each evidence requirement into a sequenced set of data retrieval tasks. For example in order to determine whether the adversary is approaching our supply base, the velocity of the adversary relative to the supply base must be determined. The data retrieval tasks are collected into itineraries for software agents. These agents are then generated and dispatched to the appropriate distributed data sources. These data sources may be the high refresh rate outputs from level 1 fusion, the structured Joint Common Database (JCDB) or Military Intelligence Database (MIDB) or the unstructured news feeds. In the case of the unstructured data sources, the agents will likely request specific searches from evidence extraction and link discovery services.

Once the agents have collected the appropriate data, SAGE takes the data and combines it into evidence. This evidence combination process varies greatly in complexity. In order to reason about the range of a SCUD, the agent may just have had to return the value of the range field for the SCUD from the JCDB. In order to reason about whether adversary tanks are approaching a blue force supply base, historical values of the tanks' position and velocities will be required as will the position of the supply base and any possible avenues of attack the adversary could follow. A further complexity is that evidence returned to AII by SAGE must have an associated probability value ranging from 0 to 1. Thus, SAGE must also estimate the probabilities associated with each piece of evidence. The second input to the adversary intent inferencing (AII) model is analyst feedback. Feedback plays a critical role and effectively updating the intent model. Feedback from the analysts in adversarial intent must be inherently uncertain. This adds an additional level of criticality to the explanation component. In particular, the intent model manages and maintains significantly more knowledge concerning the adversary than can be cognitively handled by a human analyst. Thus, by providing an explanation and even an exploration facility to the human analysts, we "open" up the intent model for complete inspection by the analysts in as organized manner as possible. In essence, we leverage the uncertainties in the analysts' inferences in order to better adapt the intent model to cover larger contingencies and increase robustness.

Conclusions

We have outlined our assessment of the best approach to addressing adversary intent inferencing based on current research and our own expertise. We are fully aware of the fact that adversary intent inferencing is a highly complex problem in which even experts do not agree on many of the fundamental issues. Currently, there are factors that we cannot concretely and precisely address but hope to do so as our project progresses. For example, we realize that with regard to observables, both user intent and adversary intent domains must determine what types / kinds of observables need to be captured for effective intent inferencing. In the user intent domain, we can assume that all observables are available and are precise. In the adversary intent domain, however, observables may not be completely obtainable or even reliable due the fog and friction of war and to deception and subterfuge on the part on the adversary. Our work is in the definition phase and this paper reflects both our past experience and current plans. In the near future we will engage with various end-user groups in the Air Force intelligence community to build a preliminary adversary intent model and to identify the information sources to be accessed. We will develop a preliminary prototype of the adversary intent inferencing model by the end of 2002.

Acknowledgements

The research presented in this paper would not be possible without the involvement of our research partners, Lockheed Martin Advance Technology Laboratories and specifically Sergio Gigli and Axel Anaruk. We thank AFRL's Information Institute for providing guidance and subject matter expertise and to AFRL's Human Effectiveness Directorate (2d Lt Sabina Noll) for providing a workshop forum for discussing the theory and requirements for adversary decision modeling.

References

Behler, R. Maj Gen 2001, "Homeland Information: AOC Can Coordinate U.S. Terror Defense," *Defense News*, 13.

Bell, B., Franke, J., and Mendenhall, H. 2000, "Leveraging Task Models for Team Intent Inference," *Proceedings of the International Conference on Artificial Intelligence.*

Bell, B., Santos Jr., E., and Brown, S. M. 2002, "Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion", *Proceedings of the 11th Annual Computer Generated Force and Behavioral Representation Conference.*

Fayette, D. F. 2001, "Effects-Based Operations: Application of new concepts, tactics, and software tools support the Air Force vision for effects-based operations", *Air Force Research Laboratory Technology Horizons*, IF-00-15.

Franke, J., Brown, S. M., Bell, B., and Mendenhall, H. 2000, "Enhancing Teamwork Through Team-Level Intent Inference," *Proceedings of the International Conference on Artificial Intelligence.*

Geddes, N. 1986, "The Use of Individual Differences in Inferring Human Operator Intentions," *Proceedings of the Second Annual Aerospace Applications of Artificial Intelligence Conference.*

Hofmann, M., Chacon, D., Mayer, G., and Whitebread, K. 2001, "CAST Agents: Network-Centric Fires Unleashed." *2001 National Fire Control Symposium: Session: Automation of the Kill Chain*, Lihue, Hawaii.

Joint Publication 3-13 1998, *Joint Doctrine for Information Operations*, Department of Defense.

Joint Publication 2-01.3 2000, *Joint Tactics, Techniques, and Procedures for Joint Intelligence Preparation of the Battlespace*, Department of Defense.

Llinas, J., Drury, C., Bialas, W., and Chen, A.C. 1998, "Studies and Analyses of Vulnerabilities in Aided Adversarial Decision Making," AFRL-HE-WP-TR-1998-0099.

Llinas, J., Drury, C., Jian, J. Y., Bisantz, A., and Younho Seong, Y. 1999, "Studies and Analyses of Aided Adversarial Decision Making Phase 2: Research on Human Trust in Automation," AFRL-HE-WP-TR-1999-0216.

McCrabb, M., Concept of Operations for Effects-Based Operations 2000, Draft paper for AFRL/IFTB, Version 2.0.

McGrath, S., Chacon, D., and Whitebread, K. 2000, "Intelligent Mobile Agents in the Military Domain." *Proceedings of the Fourth International Conference on Autonomous Agents 2000.*

Pearl, J. 1988, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann.

Saba, M.G., and Santos Jr., E. 2000, "The Multi-Agent Distributed Goal Satisfaction System," *Proceedings of the International ICSC Symposium on Multi-Agents and Mobile Agents in Virtual Organizations and E-Commerce (MAMA 2000)*, 389-394, Wollongong, Australia.

See, J. and Kuperman, G. 1997, "Information Warfare: Evaluation of Operator Information Processing Models," AFRL-HE-WP-TR-1997-0166.

Whitebread, K. and Jameson, S. 1995, "Information Discovery in High-Volume, Frequently Changing Data," *IEEE Expert Journal – Intelligent Systems and Applications.*

4.3 Adversarial Inferencing For Generating Dynamic Adversary Behavior

Joshua M. Surmana,c, Robert G. Hillmana, Dr. Eugene Santos Jr.b

[a] AFRL, Information Directorate, Advanced Computer Architectures Branch.
[b] University of Connecticut, Dept of Computer Science & Engineering
[c] University of Buffalo, Graduate Student, Dept of Mathematics

ABSTRACT

In the current world environment, the rapidly changing dynamics of organizational adversaries are increasing the difficulty for Military Analysts and Planners to accurately predict potential actions. As an integral part of the planning process, we need to assess our planning strategies against the range of potential adversarial actions. This dynamic world environment has established a necessity to develop tools to assist in establishing hypotheses for future adversary actions. Our research investigated the feasibility to utilize an adversarial tool as the core element within a predictive simulation to establish emergent adversarial behavior. It is our desire to use this intelligent adversary to generate alternative futures in performing Course of Action (COA) analysis. Such a system will allow planners to gauge and evaluate the effectiveness of alternative plans under varying actions and reactions. This research focuses on one of many possible techniques required to address the technical challenge of generating intelligent adversary behaviors. This development activity addresses two research components. First, establish an environment in which to perform the feasibility experiment and analysis. The proof of concept performed to analyze and assess this feasibility of utilizing an adversarial inferencing system to provide emergent adversary behavior is discussed. Second, determine if the appropriate interfaces can be reasonably established to provide integration with an existing force structure simulation framework. The authors also describe the envisioned simulation system and the software development performed to extend the inferencing engine and system interface toward that goal. The experimental results of observing emergent adversary behavior by applying the simulated COAs to the adversary model will be discussed. The research addresses numerous technological challenges in developing the necessary methodologies and tools for a software-based COA analysis framework utilizing intelligent adversarial intent.

Keywords: Predictive Battlespace Simulation, Adversary Behavior, Inferred Behavior, Course of Action Analysis

INTRODUCTION

Military planning systems are needed to anticipate and respond in real time to adversarial intentions with counter actions. We are faced with complex technical challenges in developing automated processes to derive hypotheses about future actions. Combat operations are conducted in the presence of uncertainty related to the disposition and intent of enemy forces. It is virtually impossible to identify or predict the specific actions an adversary might pursue. Our research interest is to develop techniques to assess specific planned courses of actions against potential adversary counter actions. Utilizing High Performance Computer (HPC) Clusters, multiple force structure simulations can be executed in parallel to concurrently evaluate the hypothesis of assessing a given COA against a range of adversary actions and belief systems. The desired goal is to establish a predictive means to evaluate the COA for critical elements related to execution

and timing as well as overall effectiveness in the presence a range of adversary counter action that may emerge.

Conventional wargaming simulations typically execute a pre-scripted sequence of events for an adversary independent of the opposing force, commonly referred to as the blue force. Dynamic adversary actions are generated to define the operational level behavior of the adversary in response to the execution of the blue force within the simulation. Multiple adversarial models with varying belief systems will be capable of automatically posing different actions and counter actions. In this manner, the blue force plans being established during the planning process are capable of being assessed against a wide range of potential adversarial actions and forces.

ADVERSARY INTENT INFERENCING

The Adversary Intent Inferencing (AII) System [1] framework is utilized to infer hypotheses from which the dynamic behavior for our adversarial behavior is generated. The AII system is a Bayesian Net Computational System that can be utilized in various ways depending on the specific application. For the application of generating adversary behavior, observations of blue actions (beliefs related to movements, activities, and capabilities) are applied to the bayesian net in the presence of given adversary actions and reactions to automatically generate hypotheses about the adversary's future actions and intent. The system requires information about our current actions and inferences about the adversary's future reactions and motivations. The AII system attempts to infer "to what extent does motivation X account for the set of observations Y" and establish predictive rankings for probabilities of 'how likely is future action Z given motivation X".

The core of the AII system carries out its computations based on Bayesian belief networks. The random variables within such networks represent objects or events and the directed arcs between networks represent direct dependencies between the variables. Each dependency is given a probability, and these probabilities are the basis of AII's computations. The networks used by AII, which represent the adversary's reasoning and action processes, contain four types of variables:

Random Variable Type	Examples
Beliefs (B) about blue force	Observations About Blue's Positions, Stereotypes About Blue's Values Thoughts About Blue's Strategy, Any Beliefs About Blue Force That May Affect The Adversary's Reasoning.
Axioms (X) that the adversary holds regarding itself ranging from capabilities to beliefs	Radar Is Operational That The Adversary Believes Itself To Be Superior At Ground Based Conflicts
Goals (G) that the adversary may wish to accomplish	Capturing Territory Preserving Its Missile Launchers
Actions (A) the adversary may take	Air Strikes, Fortifying A Position, Concealing Forces

These four random variable types are arranged into two networks that represent a single adversary. The first half of the computations is carried out on the rationale network. This

network contains all of the Belief (B), Axiom(X), and Goal (G) variables, as well as any Action (A) variables which have goals as inputs. This network is used to infer what short and long term goals the adversary may have. Once the goals are determined, the action network is used to reason on what the most likely actions will be that the adversary may carry out. The action net contains the entire set of Action (A) variables and any Goal (G) variables that could be considered as actions. As a rule, Belief variables are independent

and serve as inputs to Axioms or Goals. Axioms have Beliefs as inputs and serve as inputs to Goals and other Axioms. Goals have Axioms and Beliefs as inputs and serve as inputs to Actions or other Goals. Actions have only Goals as inputs and can only be inputs to other Actions.
AII uses these networks to establish and quantify the dependencies between these variables. A user enters in observations about the states of different variables and AII calculates the effect this evidence has on the two networks as defined below:

First, a process called belief updating [2] is performed on the rationale network to determine probabilities for the remaining random variables. In this process, if it is given from evidence that the probability of random variable A being true is the quantity *x*, or $P(A=T) = x$, then the goal is to find the probabilities of all the states of the rest of the variables in the rationale network. For example, to find the probability that random variable B is true, or $P(B=T)$, with this evidence, then we must determine several sums over the possible states of the network. In the following equations, the sum over *RV States A=T, B=T* is the sum over all possible combinations of states of random variables with the condition that A=T, B=T. $P(R \mid parents(R))$ represents the conditional probability of a state of random variable R, given the state of R's parents. If R has no parents, then $P(R \mid parents(R)) = P(R)$.

$$\sigma_T = \left(\sum_{\substack{RVStates \\ A=T,B=T}} x \prod_{R=rv} P(R \mid parents(R))\right) + \left(\sum_{\substack{RVStates \\ A=F,B=T}} (1-x) \prod_{R=rv} P(R \mid parents(R))\right)$$

$$\sigma_F = \left(\sum_{\substack{RVStates \\ A=T,B=F}} x \prod_{R=rv} P(R \mid parents(R))\right) + \left(\sum_{\substack{RVStates \\ A=F,B=F}} (1-x) \prod_{R=rv} P(R \mid parents(R))\right)$$

Once these sums of products are determined, $P(B=T) = \dfrac{\sigma_T}{\sigma_T + \sigma_F}$.

In this way, a probability is calculated for every possible state for the remaining random variables in the rationale network. As mentioned earlier, both the rationale and the action network contain all of the Goal nodes. When belief updating on the rationale network is completed, all of these Goals have probabilities associated with their respective states. This data is passed to the action network as evidence, and belief updating is then performed on the action network given this evidence.

After this second belief updating process, every random variable has been given probabilities for its states. AII examines the Actions and the Goals; the Actions are ranked according from most probable to least probable and displayed, and if states of the Goals meet certain thresholds, theses Goals are added to the goal list displayed. Also, Goals on the goal list will be added as evidence in the next iteration.

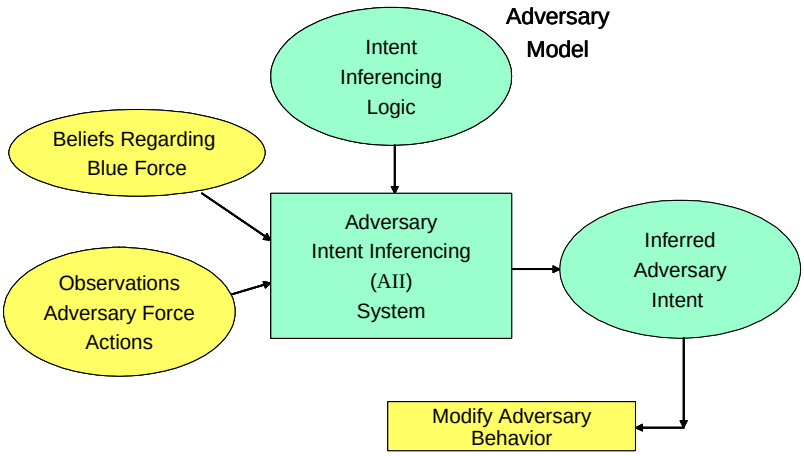


Figure 2. AII System Information Flow

The research performed was to develop and assess utilization of the AII system as an integral part of our force structure simulation test-bed providing the emergent adversary actions. The information flow between the inferencing engine within the AII system in relation to a simulated force structure simulation is illustrated in Figure 1. The requirement of updating the Bayesian model is addressed by providing data (observations) as the simulation unfolds; the inferencing requirements are achieved by creating Bayesian models to define the descriptive probabilities based on our belief system of the adversary.

CHALLENGES

The power of the inferencing engine of AII provides a great opportunity to improve predictive battlespace simulations. However, there are several challenges to investigate:

- Can we effectively create the Bayesian belief networks that will model selected adversary actions?
- Can we efficiently interface the Inference System with a Simulation framework for generating observational data?
- Will simulating planned courses of actions derive emergent behavior from the AII system?
- Can a generic Bayesian network be instantiated with alternate probabilities to establish new belief models?
- How do we devise an experiment to test and evaluate our hypothesis for emergent behavior?

Sample adversary models for demonstration purposes were included as part of the inferencing engine. However, new models specific to our application were needed for concept analysis. Our first task, therefore, was to create the processes and methodology to construct new adversary belief networks in order to effectively study the utilization of AII. The AII system requires the following three text files to encapsulate the adversarial model:

1. Rationale Network – list all of the random variables and give the probability tables for each of the conditional dependencies represented in the motivational Bayesian network.
2. Action Network – list all of the random variables and give the probability tables for each of the conditional dependencies represented in the actions Bayesian network.

3. Adversarial Model File – Defines the reference relationships between the Action Network and the Rational Network.

SYSTEM INTERFACE

A Graphical application is best suited for creating these files since the files essentially represent directed graphs. This application also is effective in managing the size and complexity of the truth tables and nodal list being generated. A Java editor "JavaBayes version 0.346" [3] was selected because it suited our generation needs and with the Java source available it could optimize for creating AII-compatible networks. This application can create and manipulate random variable nodes and dependencies between nodes through a user-friendly graphical interface within a Java window. The probability tables for the variables can also be set up with ease. For efficient viewing and organization of the AII nets, the Java code was extended so that all the node name identifiers are color-coded based on the associated AII variable type. The AII model networks created can be saved in Bayesian Interchange Format (BIF) which is easily converted to the format used by AII. Thus, an effective method for creating, organizing, viewing and saving Bayesian networks for this specific inferencing application was established.

The next task was to modify and streamline AII's I/O process and establish a semantic interface for the AII engine to interact with a wargaming simulator. In order to enter data and observations using the default AII system, a user has to navigate menus with numerical selections using a standard keyboard interface. To simplify this process, the numbered menus were bypassed and an input parser added. This way, a user can enter any observations in a single step. The command line parser was constructed with three fields: a reference to a random variable, a true or false flag, and the decimal value for the variable between 0 and 1. Now variable updates can be entered easily and without the constraint of the menu interface.

The next step was to establish a means for AII to accept and process input data with other software programs and applications. In this way, AII can receive evidence from data-collecting agents, and then output the adversary intent to other applications such as a wargaming simulator. A TCP/IP socket interface was integrated into AII as the input mechanism to satisfy our system input requirements. The same mechanism can also be used to send the inferenced results to external applications, allowing the results of the engine to provide the emergent behavior to the simulation framework. A second benefit of establishing the new interface system was an effective means to perform the feasibility analysis. The process of analyzing input data sets and adversary models to validate emergent behavior is discussed in the next section.

As stated before, the original code simply used standard I/O for all interactions with the user interface. The evidence list is updated and appended to the output stream whenever an observation is entered. The goal list and ranked action list are output in the same manner at the end of every time step. To achieve our display requirement, a Java Native Interface (JNI)[] was developed as the GUI display mechanism that captures the inferred data updates. The JNI that was utilized allows non-Java programs such as the AII C++ program to run Java methods in a virtual machine. The Java GUI window contains three scrolling panes for the three AII output lists: evidence, goals, and ranked actions. The Java window updates the output rather than appending the data and displays the information into an easily readable format.

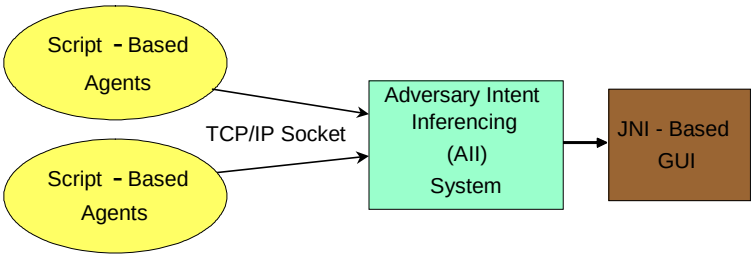


Figure 3. System Interfaces

Figure 2 depicts the interfaces established by these modifications to AII. Agents are capable of communicating to the AII system input through the TCP/IP socket. The system infers the adversary's ranking for the actions and goals then outputs the results to a Java display window utilizing JNI.

ADVERSARY MODELS

The core of AII's engine performs all of its calculations using the values and relationships from the Bayesian networks. The importance of accurately encapsulating the adversary within these networks is, therefore, very clear. However, the size of the networks expands dramatically as each random variable is added. Since building such large networks is challenging and time-consuming, a goal set was to create a generic model that could be modified to portray different adversaries. By determining which factors affect most forces, the same random variables could be preserved from model to model. This way, when a new adversary is to be rendered, portions of the Bayesian Model could be reused with the probability tables modified.

In the first attempt at such a general model, the included random variables accounted for aspects, such as weapon capability, tactical strikes verses strategic strikes, and responses to the presence of blue forces in any of the four cardinal directions. Two instances of this model with different belief systems (characteristics) were created, referenced as adversary A and B. Adversary A possessed a competent air force, a smaller ground force and had Weapons of Mass Destruction (WMD). Adversary B was characterized having a powerful ground force, but little air power and no WMD capability. Adversary A was characterized by a confidence in its higher technology, whereas adversary B was characterized by religious fanaticism. It would be expected; therefore, that adversary A would respond to blue force actions by air counterstrikes and potentially even employing their WMD, whereas adversary B would respond best through ground actions. The more interesting aspect would be related to the sequencing, timing, and order of the actions that were generated.

To test AII against these hypotheses about the adversaries' behavior, two different sets of observations were created that were fed into the AII system. The blue force observation was divided into four subsets of time each representing 30 minutes. Therefore for the purpose of this analysis each inferenced time step is 30 minutes. In the first input set, data that would be indicative of a strategic bombing campaign, including deploying sea forces, launching cruise missiles, and air strikes for strategic targets. The second input set was designed to portray a blue force land invasion aimed at military targets; included indications that the blue force was present to the North and making a ground assault from the West possible, blue force ISR assets deployed, and blue conducting tactical air strikes. An important point to realize is that the differences in the inferred hypotheses for future adversary actions is based solely on difference in the belief systems rather that any changes in the Bayesian structure. It is expected that changes in the Bayesian

structure would derive additional action variants. By modifying only the belief system, a more difficult test set has been established for deriving different action sets from the two adversaries.

	Adversary A	Adversary B
COA Input set 1	Deliver Ultimatum Launch Air Attack Send Forces South Arm Weapons Of Mass Destruction Launch Weapons Of Mass Destruction	Launch Ground Assault Send Forces South Enemy Recon Probing Forces Cross Border Deploy Forces In Civilian Areas
COA Input set 2	Deploy Forces In Civilian Areas Deliver Ultimatum Deploy Forces Along Border Arm Weapons Of Mass Destruction Conceal Assets Launch Weapons Of Mass Destruction	Deploy Forces In Civilian Areas Launch Ground Assault Send Forces West Send Forces North Enemy Recon Probing Forces Cross Border Deploy Forces Along Border Conceal Assets

Table 1. Sample Results Matrix

Table 1 lists some of the highly ranked actions and differences resulting from the input sets which, as anticipated, exploited the differences in the adversary belief systems. The Graphs in figure 3 and figure 4 contain plots of the adversary actions as enumerations and the associated rankings that AII generated. A higher number suggest that those actions are being pursued. The first plot compares the actions of adversary A as the first blue force COA is applied. The graph illustrates the dynamics in the anticipated behavior in responding to the blue force at each time interval. Figure 4 illustrates the divergence between the two adversaries' action hypotheses at a single time step. A similar divergence exists at each time step.

Figure 3. Dynamics of Action behaviors Adversaries

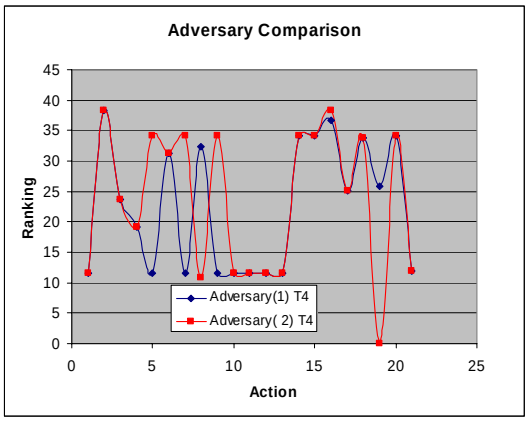
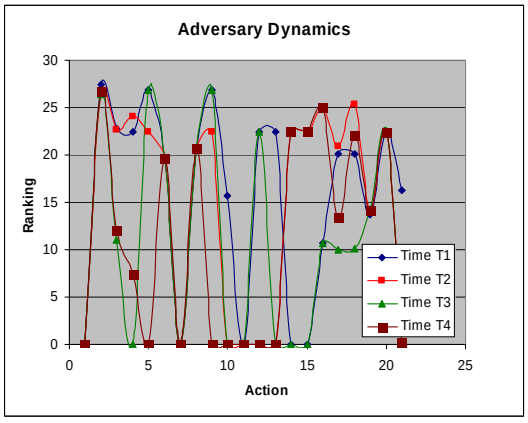


Figure 4. Action differences between



In both COA input sets, the possibilities existed that an adversary could threaten and employed their WMD, or launch an attack with its ground forces. The differences in the input COA sets were quite apparent, however. In input set 1, since the blue force focus was on strategic goals and not tactical targets, both adversaries seemed probable to attack with their strongest forces. In input COA set 2, the tactical strikes against military targets made by blue force made it likely that both adversaries would try to conceal their forces, perhaps even in civilian areas, and the blue

ground assault would likely result in resistance forces deployed along borders. As hypothesized, the differences in the belief probabilities made a noticeable difference in the behaviors of the adversaries created with the general model. Their behaviors seemed reasonable and characteristic of the strengths and weaknesses portrayed in the models.

FUTURE WORK

Once the effectiveness of general models was confirmed, the next focus of the project is to establish a connection between the pre-existing wargaming simulator and inferencing engine within AII. An agent interface is being developed to act as the gateway providing the blue force observations to AII.

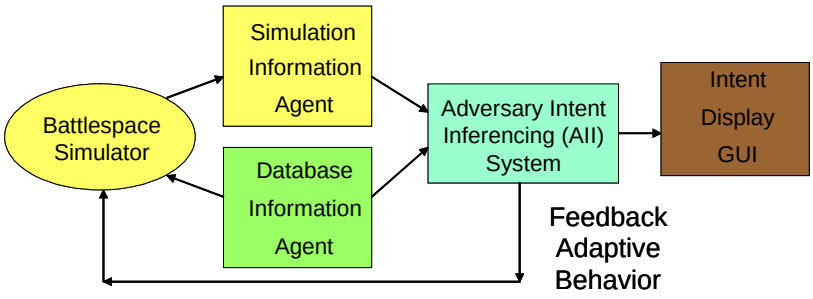


Figure 5. Integrated AII and Wargaming Environment

Figure 5 illustrates the complete environment planned for the simulation environment. Two agents are necessary to provide the blue force observations. One agent provides the observations as discussed above and a second agent is required to allow the initial states to be defined. Some portion of the pre-deployment or staging performed by the blue force needs to be extracted from the plans database in order for the adversary to establish its' initial actions.

Since the eventual goal is for AII to drive the actions and provide the rationale behind the simulated adversaries, the variables used within the model will have to correspond to actions and behaviors within the simulator. For this reason, the simulator requires AII to utilize operational level models. Development of such high level models is ongoing. While the North/South/East/West directions of the previously discussed model were, in fact, at too low a conceptual level, variables such as "Fortify City A." will be included, as the simulator is capable of carrying out such an action. The NSEW approach, while useful for testing purposes, is not supported by the wargaming simulator.

Finally, changes are being made to the next version of AII to help in the knowledge acquisition process. The intent is to provide a capability that allows a user to design rules with "blank" variables so that different random variables can be instantiated to create the network from these rules. Interconnection guidelines will be defined for these rule templates to satisfy the constraints on how random variables can be connected. This rule based structure is the first phase in building a knowledge acquisition interface to aid in the generation of the adversarial models.

CONCLUSION

The Bayesian net editor allowed us to create new adversarial models corresponding to the semantic model structure defined by the AII tool. These models were used to test AII capabilities and effectiveness at developing the hypotheses of adversary actions. The modified I/O system with socket laid the groundwork for interfacing with external simulations and agent interfaces, and the JNI code allowed us to utilize the I/O abilities of Java to format our output in a useful manner. The first phase of the project successfully concludes verifying both the ability to readily interface the AII tool to multiple input and output sources and provide the desired emergent actions to drive the adversary behavior. The next stage of establishing the integrated environment is under development. The top technical challenge to the successful integration of the framework will be in creating the semantic interface that needs to be established between AII and the Wargaming simulation.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the significant contributions to this project made by Mr. Sergio Gigli (Lockheed Martin, Advanced Technology Laboratories) and Mr. Martin J. Walter and Mr. James P. Hanna (Air Force Research Laboratory, Information Directorate).

REFERENCES

1. Bell, B. and Santos, E., Jr., "Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion." Proceedings of the 11th Conference on Computer Generated Forces and Behavioral Representation, Orlando, FL, 2002.

2. Pearl, Judea, "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference", Morgan Kaufmann, San Mateo, CA 1988

3. Cozman, F., "Generalizing Variable Elimination in Bayesian Networks." Workshop on Probabilistic Reasoning in Artificial Intelligence, Atibaia, Brazil, November 20, 2000.

In *Proceedings of the SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, Orlando, FL, 2003

4.4 A Cognitive Architecture for Adversary Intent Inferencing: Structure of Knowledge and Computation

Eugene Santos Jr.
University of Connecticut, Dept of Computer Science & Engineering, Storrs, CT 06269
eugene@cse.uconn.edu

ABSTRACT

Existing target-based and objectives-based ("strategy-to-task") approaches to mission planning do not explicitly address the adversary's decision-making processes. Obviously, the adversary's courses of action (COA) are influenced in a cause-and-effect manner by actions taken by friendly forces. Given the iterative/interleaved nature of actions taken by enemy and friendly forces, mission planning must clearly take adversarial decision making into account especially during concurrent mission planning and execution. Currently, adversarial behaviour with regards to cause-and-effect are difficult to account for within the framework of existing planning approaches. This paper describes a cognitive architecture for computationally modeling, predicting, and explaining adversarial behaviors and COAs and proposes an integrated framework for mission planning. Our framework fits naturally within the Effects-Based Operations (EBO) approach to mission planning.

Keywords: Adversary Behavior, Inferred Behavior, Course of Action Prediction, Course of Action Analysis, Cognitive Architecture, Bayesian Networks, Reasoning, Computation, Enemy Course of Action

INTRODUCTION

Currently, target-based and objectives-based ("strategy-to-task") approaches to planning do not explicitly address the adversary's decision-making processes [6]. In particular, the adversary's behaviors and courses of action (COA) are obviously influenced in a cause-and-effect manner which are actually difficult to account for within the framework of existing planning approaches. Furthermore, friendly COAs directly impact future adversary actions. Thus, to properly take into account adversarial decision-making and behavior in mission planning requires the capability to model adversaries in order to (1) predict enemy COAs, (2) infer adversarial intent, and (3) explain the rationale behind adversarial behaviour and intent. Such a model must capture a variety of attributes including effects of friendly and enemy behavior/action over time with regards to the current situation as well as soft factors such as enemy will, political system, and system of beliefs.

The foundations supporting the development of such adversary-aware mission planning systems are emerging from USAF-sponsored research. This approach, termed effects-based operations (EBO), is the best candidate to serve as the basis of the operations model we require [17]. Basically stated, EBO is "an approach that…explicitly seeks to understand, trace, and anticipate direct and indirect effects of a specific action…on an adversary's course of action [13]." EBO is framed with respect to outcomes produced (and/or predicted to be produced) in the battlespace. EBO inherently addresses an adversary as a system. The notion of "effect" is predicated upon the

presumption that there is an object of reference (specifically one systemically organized), namely the adversary, whose state(s) can be identified and influenced through prospective courses of action. EBO planning is predicated on a coherent model of the state(s) and dynamics of the adversary system(s). At the center of the EBO concept is the idea that effective friendly COA planning can and should be framed with respect to effects to be induced in an adversary system.

The key to effects-based operations revolves around determining how an adversary should / can / could react to system perturbations resulting from actions on the battlefield from our own forces [17]. One of the greatest technological challenges for the EBO approach is that of adversarial decision modeling. While EBO's overall goal is to model the enemy in its entirety (stated as "enemy-as-a-system" in the EBO CONOPs and including the physical, data, cognitive, and social aspects of the battlespace centers of gravity and the dependency linkages between them), we believe that a necessary starting point is to model an adversary commander's intent. Intent inference involves deducing an individual's goals based on observations of that individual's actions [15]. These inferences can then be passed to an application for generation of advice, definition of future information requirements, proactive aiding, or a host of other benefits. Furthermore, the success of adversary intent inferencing addresses a key technological barrier of EBO—that of the human element's impact in EBO. Once adversary intent is suitably modeled and captured, we can then compose these individual adversary commander's intent models into larger collectives using team intent modeling to address the general problem of the "enemy-as-a-system."

Our goal in this paper is to present a cognitive architecture for adversarial modelling and provide a computational framework for adversary intent inference.

BACKGROUND AND APPROACH

Intent inference involves deducing an individual's goals based on observations of that individual's actions [15]. In automated intent inference, this process is typically implemented through one or more behavioral models that have been constructed and optimized for the individual's behavior patterns. In an automated intent inference system, data representing observations of an individual, the individual's actions, or the individual's environment (collectively called observables) are collected and delivered to the model(s), which match the observables against patterns of behavior and derive inferred intent from those patterns. These inferences can then be passed to an application for generation of advice, definition of future information requirements, proactive aiding, or a host of other benefits.

There are three major functions of intent inference [14]: *Descriptive* intent inference provides insight into the motivations behind actions that have just occurred. *Predictive* intent inference can anticipate future actions given the individual's inferred goals. *Diagnostic* intent inference arises from a targeted combination of predictive and descriptive models, which compares previous predictions against current knowledge. This works to reveal discrepancies in either the models (supporting a model adaptation and learning function) or the real world (identifying mistakes made by the individual).

Each function of intent inference can be dissected into three informational components [23, 24, 25]: The first, *interests and focus*, captures at a high level the direction of the individual's attention. The second, *actions and preferences*, describes the activities that can be used to carry out the goals that currently hold the individual's attention, with a focus on how the individual tends to carry them out. The third, *knowledge and reasoning*, provides insight into the deeper motivations behind the goals upon which the individual is focused and illuminates connections

between goals. In other words, the first component captures what the individual is doing, the second captures how the individual might do it, and the third infers why the individual is doing it.

Applying the principles of modeling the what/how/why of individual intent, we see that our approach naturally integrates into the major themes of EBO. While the field of individual intent inference has historically focused on better improving the human-system interface, we contend that there is a natural isomorphism between our own prior work in the field of user and team intent inference [2, 3, 4, 5, 14, 22, 23, 24, 25] and the domain of adversary intent inference [1, 6, 29]. While the operational world surrounding an intent inference application would be very different, the inner mechanisms of intent inference map directly between domains.

While observables in the user intent domain stem from data collected from human use of systems, observables in the adversary intent domain take the form of tactical information derived from intelligence databases, observations of the tactical environment, and input from online human experts. In place of window events, keystrokes, and mouse movements, our system in the adversary intent domain uses information about adversary location, movements, and activities to drive its inference. In place of computer state, analyses of information queries, and the content of user dialogue with team members, our system bases inferences on facts about the local terrain, regional weather, and the salient political climate.

Likewise, tactical goals will replace computer operational goals in the results of our intent inference. Descriptive intent inference in this case would result in identification of an adversary force's objectives and, given models of tactical reasoning, could recommend appropriate reactions. Predictive intent inference would indicate expected activity by the adversary and explain the reasons behind that activity. Diagnostic intent inference could produce alerts of attempted subterfuge or uncover missteps on the part of the adversary.

Similarly, intent inference of echelons of adversary forces provides advantages reminiscent of those that arise from team intent inference. Identification of the goals of one adversary group can be used as a discriminator in identifying the goals of other subsets of the adversary. Intent inference across groups of the adversary could also result in the discovery of breakdowns among those groups, knowledge that can be used to our tactical advantage. We now present the cognitive architecture that forms the basis for our adversarial modeling.

ADVERSARY INTENT INFERENCING MODEL

To achieve adversarial intent inferencing requires the ability to (1) fuse information (observables) from sensors and intelligence sources regarding the adversary, (2) infer adversary intent and goals, and (3) predict adversary courses of action (COA). In total, adversary intent inferencing (AII) provides these three key functions while also taking into consideration a number of utility issues:

- AII must be able to explain the basis of its predictions; why is the adversary pursuing a predicted goal? What is driving the adversary to pursue these COAs? Must be able to model and take into account many factors including soft factors such as political environment, personality issues, adversarial religious beliefs, etc.
- AII must be able to adapt predictions based on history of events and observed enemy operations.
- AII must be dynamic and able to learn changes in the adversary behaviour and ultimately model pop-up adversaries; AII must be able to provide the capability for standing up models of new adversaries while avoiding the knowledge/information engineering bottleneck.

While the ultimate role and capabilities of AII still requires a great deal of long term research, from our first steps thus far, we can carefully identify capabilities that are likely achievable in the short-term for mission planning and wargaming to support Effects-based Operations (EBO), Predictive Battlespace Awareness (PBA), and Intelligent Preparation of the Battlespace (IPB).

As we discussed above, we derive our adversarial architecture from the formative components found in our user modeling approach to intent inferencing. In particular, we preserve the structure of the what/how/why model in order to provide a natural and intuitive decomposition of both the adversarial decision-making process and central knowledge-base. The benefits of such a decomposition are two-fold: First and foremost is the classic bottleneck of knowledge acquisition. Our decomposition provides a critical organizational structure in order to better capture/construct adversarial knowledge-bases/models in a manageable fashion. Secondly, with more modular components, this eases the issues of computational complexity and validation/auditability of the inferencing process.

The components of our adversary intent inferencing model, and the interactions between these components, are shown in Figure 4. The three core components that comprise our architecture and functions are as follows:

- **Goals**: Probabilistically prioritized short- and long-term goals list, representing adversary intents, objectives or foci
- **Rationale**: A probabilistic network, representing the influences of the adversary's beliefs, both about themselves and about us, on their goals and on certain high level actions associated with those goals
- **Actions**: A probabilistic network, representing the detailed relationships between adversary goals and the actions they are likely to perform to realize those goals

The goal component captures what the adversary is doing, the action component captures how the adversary might do it, and the rationale component infers why the individual is doing it. Due to the inherent uncertainty involved in adversary course of action prediction, we use Bayesian networks [18] as the main knowledge representation for the rationale and action networks. Each random variable (RV) involved in the Bayesian networks is classified into one of four classes: axioms, beliefs, goals and actions. Each RV class is described below:

(a) **Adversary axioms (X)** – represents the underlying beliefs of the adversary about themselves (vs. beliefs about our forces). This can range from an adversary's beliefs about his or her own capabilities to modeling a fanatic's belief of invulnerability. Axioms typically serve as inputs or explanations to the other RVs such as adversary goals

(b) **Adversary beliefs (B)** – represents the adversary's beliefs regarding our forces (e.g., an adversary may believe that the United States is on a crusade against them or that the United States is not carpet-bombing territory)

(c) **Adversary goals (G)** – represents the goals or desired end-states of the adversary. These goals are defined as either short-term or long-term in a goals list. Further we partition goals into two types: abstract and concrete. Abstract goals are those that cannot be executed (e.g., preserving launchers, damage US world opinion, defeating US foreign policy).

(d) **Adversary actions (A)** – represents the actions of the adversary that can typically be observed by friendly forces.

These four random variable types are arranged in the two networks: rationale network and action network. The rationale network contains all of the Belief (B), Axiom(X), and Goal (G) variables, as well as any Action (A) variables which have goals as inputs. This network is used to infer what short and long term goals the adversary may have. Once the goals are determined, the action network is used to reason on what the most likely actions will be that the adversary may

carry out. The action net contains the entire set of Action (A) variables and any Goal (G) variables that could be considered as actions. As a rule, Belief variables are independent and serve as inputs to Axioms or Goals. Axioms have Beliefs as inputs and serve as inputs to Goals and other Axioms. Goals have Axioms and Beliefs as inputs and serve as inputs to Actions or other Goals. Actions have only Goals as inputs and can only be inputs to other Actions. There are additional rules for structuring the relationships between random variables that are not directly described here due to space limitations but can be seen in the example networks. Figure 5 depicts a rationale network and an action network.

The AII process (as shown in Figure 4) works iteratively as follows:

1. Observables regarding the adversary such as actions and beliefs are set as evidence in both rationale and action networks (depicted as red nodes in figure). Also, feedback from analyst is set as evidence.
2. Current short- and long-term enemy foci from the foci lists are also set as evidence in both networks (depicted as green nodes).
3. The rationale network is then used to infer new goals which are set as evidence for the action network.
4. The action network is now used to predict adversarial actions.
5. The analyst is presented with the inferred goals and predicted actions.
6. The analyst provides feedback in terms of corrected goals and actions if desired.
7. The goals list is updated based on newly inferred goals and current strength of existing goals. If goals exceed a given threshold value, they are added to the list. If goals fall below a set threshold, they are removed. If goals in the short-term list persist beyond a given time threshold, they become long-term goals.
8. Go to step 1.

Due to space considerations for this paper, we have omitted details regarding specific functions/formulas used. The inference process on both the rationale and action networks is based on belief updating [18]. In essence, given a target random variable R and evidence set E, belief updating computes P(R|E) assuming that random variables have two states (true/false) for simplicity of discussion. In the following section, we give a brief overview of Bayesian networks in order to provide a better understanding of uncertainty modeling issues involved in our framework.

As we can see in the above process, the adversary model is capable of adapting to changes in the adversaries goals and intentions over time as reflected in the enemy foci lists. Also note that there are feedback and explanation paths within the adversary intent inference (AII) model. Feedback from a human analyst, although unlikely to be totally certain, can be extremely valuable to the AII model, correcting and extending its intent inferencing logic. Explanation capabilities are essential in order for intelligence analysts, using AII, to understand why the AII model has reached particular inferences. The analysts must be able to inspect the reasoning paths used by AII so that they can develop a level of confidence in the output of the AII model.

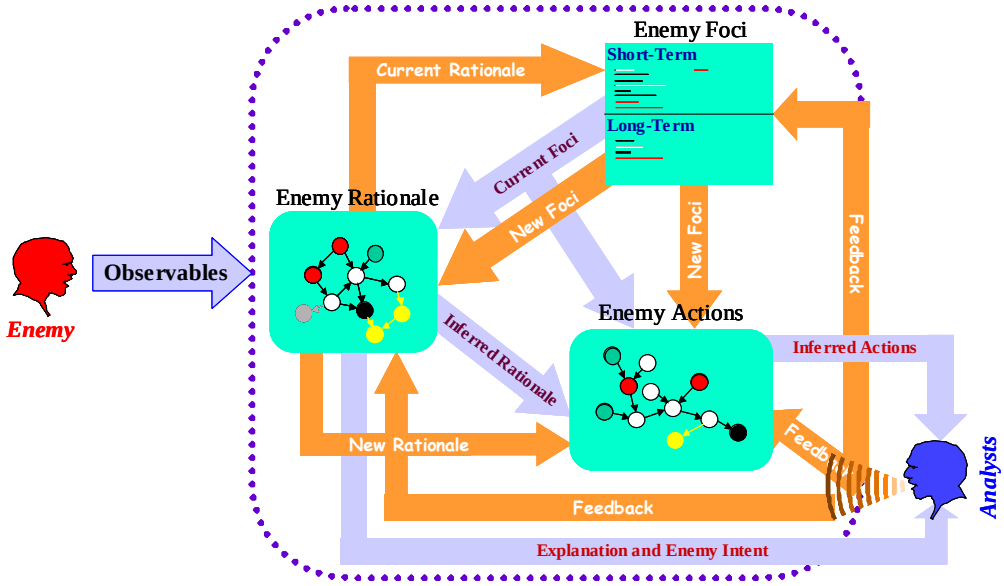


Figure 4. AII Process

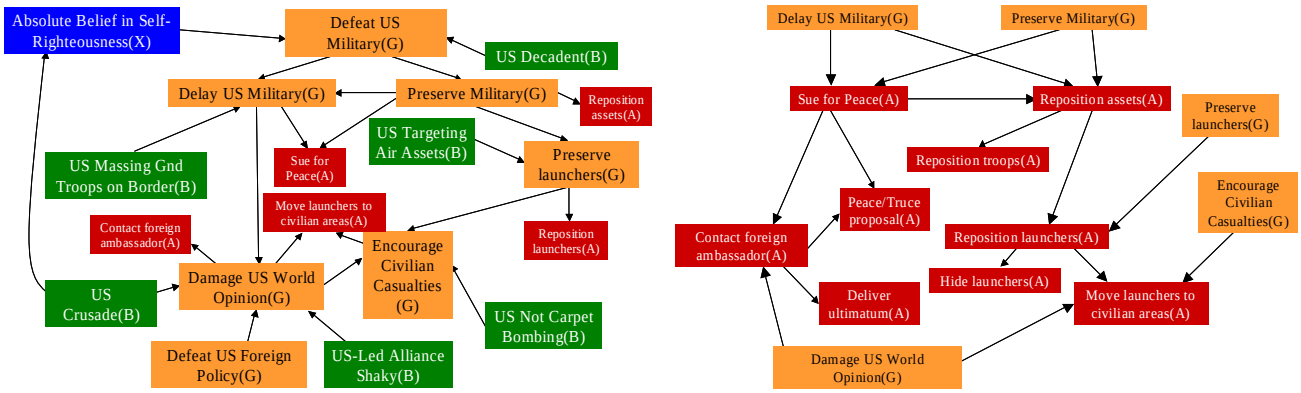


Figure 5. Rationale Network (left) and Action Network (right). The random variables are labeled according to their categories.

UNCERTAINTY

Our rationale and action networks must capture the uncertainties inherent in the adversarial model as well as the uncertainties found in the observables. In probabilistic reasoning, random variables (abbreviated, r.v.) are used to represent events and/or objects in the world. By making various instantiations to these r.v.s, we can model the current state of the world probabilistically. Thus, this will involve computing joint probabilities of the given r.v.s. Unfortunately, the task is nearly impossible without additional information concerning relationships between the r.v.s. In the worst case, we would need the probabilities of every instantiation combination which is combinatorially explosive.
On the other hand, consider the chain rule as follows:

$$P(A1, A2, A3, A4, A5) = P(A1 \mid A2, A3, A4, A5)\, P(A2 \mid A3, A4, A5)\, P(A3 \mid A4, A5)\, P(A4 \mid A5)\, P(A5).$$

Bayesian networks [18] take this process further by making the important observation that certain r.v. pairs may become uncorrelated once information concerning some other r.v.(s) is known. More precisely, we may have the following independence condition:

$$P(A \mid C1, ..., Cn, U) = P(A \mid C1, ..., Cn)$$

for some collection of r.v.s U. Intuitively, we can interpret this as saying that A is determined by C1, ..., Cn regardless of U.

Combined with the chain rule, these conditional independencies allow us to replace the terms in the chain rule with the smaller conditionals. Thus, instead of explicitly keeping the joint probabilities, all we need are smaller conditional probability tables which we can then use to compute the joint probabilities.

In Bayesian networks, these conditional dependencies are represented as a directed acyclic graph of r.v. relationships. Directed arcs between r.v.s represent direct conditional dependencies. When all the parents of a given r.v. A are instantiated, that r.v. is said to be conditionally independent of the remaining r.v.s which are not descendants of A given it's parents. (For more details on this, see d-separation in [18].

For example, let's consider the following story: Mary walks outside and finds that the street and lawn are wet. She concludes that it has just rained recently. Furthermore, she decides that she does not need to water her climbing roses. Assume that Mary used the following set of rules:

rain or sprinklers --> street = wet
rain or sprinklers --> lawn = wet
lawn = wet --> soil = moist
soil = moist --> roses = okay

We can directly transform these into a graph. Now, by considering each variable as a r.v. with possible states of {true,false}, we can construct conditional probability tables for r.v. which reflects our knowledge of the world. The joint probability of the world where the roses are okay, the soil is dry, the lawn is wet, the street is wet, the sprinklers are off and it is raining is computed as follows:

$$P(\text{sprinklers} = F, \text{rain} = T, \text{street} = \text{wet}, \text{lawn} = \text{wet}, \text{soil} = \text{dry}, \text{roses} = \text{okay}) = P(\text{roses} = \text{okay} \mid \text{soil} = \text{dry}) * P(\text{soil} = \text{dry} \mid \text{lawn} = \text{wet}) * P(\text{lawn} = \text{wet} \mid \text{rain} = T, \text{sprinklers} = F) * \& \& P(\text{street} = \text{wet} \mid \text{rain} = T, \text{sprinklers} = F) * P(\text{sprinklers} = F) * P(\text{rain} = T)$$

Substituting the appropriate numbers from the tables, we get 0.2 * 0.1 * 1.0 * 1.0 * 0.6 * 0.7 = 0.0084 as the probability of this scenario.

There are two types of computations performed with Bayesian Networks: belief updating and belief revision [18]. Belief updating concerns the computation of probabilities over random variables, while belief revision concerns finding the maximally probable global assignment.

Belief revision can be used for modeling explanatory/diagnostic tasks. Basically, some evidence or observation is given to us, and our task is to come up with a set of hypothesis that together constitute the most satisfactory explanation/interpretation of the evidence at hand. This process has also been considered abductive reasoning in one form or another [19]. More formally, if W is

the set of all r.v.s in our given Bayesian network and e is our given evidence, i.e., e represents a set of instantiations made on a subset of W, any complete instantiations to all the r.v.s in W which is consistent with e will be called an explanation or interpretation of e. Our problem is to find an explanation w* such that

$$P(w^* \mid e) = \max P(w \mid e).$$

w* is called the "most-probable explanation." Note that to compute the most-probable explanation for e, it is sufficient to determine the complete assignment consistent with e whose joint probability is maximal. In this case, P(e) is simple a constant factor. Intuitively, we can think of the non-evidence r.v.s in W as possible hypotheses for e.

Belief updating on the other hand is interested only in the marginal probabilities of a subset of r.v.s given the evidence. Typically, it is to determine the best instantiation of a single r.v. given the evidence. For example, let the evidence e be the observation that the roses are okay and the condition of our lawn be our focus. Our goal is to now determine the probability that our lawn is either wet or dry given the observation. The solution then becomes –

$$P(\text{lawn} = \text{dry} \mid \text{roses} = \text{okay}) = 0.1190$$
$$P(\text{lawn} = \text{wet} \mid \text{roses} = \text{okay}) = 0.8810$$

Although performing belief revision and updating (even approximating methods) have been shown to be NP-hard, there exist special network toplogies for which certain algorithms perform well such as polytrees [18]. Various approaches to reasoning with Bayesian Networks include A* search, stochastic simulation, integer programming, and message passing [20, 26, 28, 21, 27, 33]

WHAT'S NEXT?

Currently, the AII adapts by capturing temporal changes in adversarial activities through the short-term and long-term foci lists. While we believe that this is one of the most critical capabilities that must be provided for useful adversarial behavior prediction, additional adaptive capabilities are needed to ultimately solve problems such as pop-up adversaries. In the worst case, one of the primary difficulties with pop-up adversaries is the potential lack of knowledge or incompleteness of information available to build our networks apriori. To address this challenge, the AII must be capable of automatically updating its network models by adding or removing nodes as observations, predictions, and feedback is garnered regarding the adversary. In Figure 4, such changes are made in the feedback stage where the yellow nodes represent new additions and the gray node represents deletion. Our vision initially for achieving this adaptability focuses on two elements: (1) the identification of a need for new knowledge or removal of old/incorrect knowledge, and (2) the construction/destruction of knowledge. For (1), we can detect this situation when the recent feedbacks from the analyst are significantly different from or contradicts the AII predictions. In this case, the answers provided by AII are inconsistent either because of *incompleteness* -- insufficient information (relevant nodes in the networks) currently captured, or *incorrectness* -- there is information that is incorrectly captured. To address incompleteness, we envision a library of knowledge nuggets which are small network fragments corresponding to simple rules or templates. When additional information is needed, the library is referenced and the appropriate knowledge nuggets are obtained and introduced (like the yellow nodes) into the networks. For incorrectness, we can initiate sensitivity analysis on the current networks to identify the nodes that are the significant cause of the inconsistency. Once identified, they and nearby nodes are removed and stored back into the library as knowledge nuggets. One of the side benefits of this approach is that adversarial models can be constructed on the fly and

computational costs from inferencing can be better controlled. Also, the knowledge nuggets should be easier to formulate and validated while being applicable to various large collections of different adversaries.

Clearly, what we have just outlined so far in this paper reflects our current beliefs on the best approach to addressing adversary intent inference based on current research and our own expertise. We are quite aware of the fact that adversary intent inference is a highly complex problem in which even experts do not agree on many of the fundamental salient points. Currently, there are factors that we cannot concretely and precisely address but hope to do so as our efforts during the project provide us with more insights both from successes and failures. For example, we realize that with regard to observables, both user intent and adversary intent domains must determine what types/kinds of observables need to be captured for effective intent inference. However, for user intent, we can assume that all observables are available and are precise. For adversary intent, observables may not be completely obtainable or even reliable due the factors arising from the fog and friction of war effects to deception and subterfuge on the part on the adversary. In the next section, we present a longer term vision for properly integrating adversary intent inferencing that we believe will help address many of these issues.

THE BIG PICTURE

To recap, understanding the adversary is a long and well-known fundamental need for effective military planning and operations. A major challenge we face today is in providing the ability to explain as well as predict enemy intentions, goals, behaviours, plans, and actions and then effectively integrating this information into blue forces mission planning and execution. This is further complicated by the online nature of real-world operations in which actions taken by the blue forces will undoubtedly affect future actions and goals of the adversary. Also, the inherent uncertainty such as the fog-of-war and enemy deception constrains the information obtainable both in terms of quality and temporal availability. All of this must also take into consideration the limited resources available for information gathering. In this section, we present some initial thoughts and ideas on how-to and what it takes to properly account for the adversary.[6]

We propose a model and architecture for adversary intent inferencing and course-of-action prediction with dynamic information fusion and gathering. Our goal is to provide a unified approach that is composed of 3 major elements: (1) adversary modelling and intent inferencing, (2) adversary plan recognition and course-of-action prediction, and (3) adaptive information tuning and fusion. The first component provides the basic capability to model the adversary and explain/predict their behaviour from observations gathered by component (3). The plan recognizer and adversary COA prediction then uses the predicted behaviour from (1) and observables from (3). The results of (2) can then be directly used by blue planning systems. Our information tuning/gathering system then uses the results from (1) and (2) to retask themselves to either validate a prior observable or search for new observables that can further improve the predictions of (1) and (2). This is an iterative/online process that we will now consider in more detail.

Figure 6 presents a high-level view of the proposed Adversary Intent/COA recognizing system proposed. As shown, there are three key components: Adversary intent inferencing, case-based plan recognition, and the adaptive information system. We now present these three components in detail.

[6] The ideas presented here resulted from joint work between the author, Scott Deloach, and Michael Cox.

Adversary Intent Inferencing. The adversary intent inferencer is responsible for determining red goals including strategic and high-level tactical goals and actions.

The AII takes input from 3 sources:

1. Evidence/observables from AIS – theses may be observations straight from the battlefield, recon, sensor arrays, etc.
2. Projected Red COAs from plan recognizer – these are critical to better guiding the AII in identifying red goals allowing for more flexibility in dealing especially with soft factors without the explosion of uncertainty.
3. Analyst input – critical to merging and working with human analysts.

A complete system must encompass all 3 in order to complete the cycle of red forces analyses and for proper incorporation into blue forces planning. With regards to input 2, hard factors as opposed to soft factors can be characterized as factors that are measurable, observable, or physical in nature. Soft factors, on the other hand, include intangibles such as enemy will, political influences, personality, human behaviour, and fundamental belief systems.

Case-Based Plan Recognition. The plan recognizer is responsible for providing a key piece of information the analyst desires. That is it takes as input targeted information from the AIS (e.g., blue goals and plans along with red actions) and high-level red goals predicted from AII, and it produces as output possible red courses of actions (COAs). This output will be directed to the tuner mechanism of AIS, to AII as feedback, and to the analyst. Analysts can then make decisions (including subsequent parameter tuning of the AIS) based upon the potential red COAs.

To perform such predictions the system must reason about information concerning both red and blue, because possible responses from each will interact. One obvious interaction is the relative physical deployment locations of forces. The COAs predict employment of red forces, but this prediction also depends on current blue deployment and what red knows of it. Managing these interactions well crucially depends upon knowing both blue goals (given from AIS) and likely red goals (inferred by AII). Knowledge about goal interactions (e.g., goal conflict. See [31]. and the possession of a general theory of goal change and management [7, 8, 10, 32, 12] enables the recognizer to constrain the possible high level interpretations of events using the Meta-AQUA system [9]. Statistical methods used by the incremental case-based plan recognition system [16] are thus more tractable.

The COAs are also just a prediction, so the recognition component will directly provide AIS with future events to monitor in order to strengthen or modify such predictions. These time sensitive information requests are very similar to rationale-based planning monitors [30], but they are spawned in response to inferred red plans rather than known blue plans. Finally, particular blue actions interact with other blue intentions. Therefore AIS will be directed to monitor a select key set of future blue events and report to the plan recognition component.

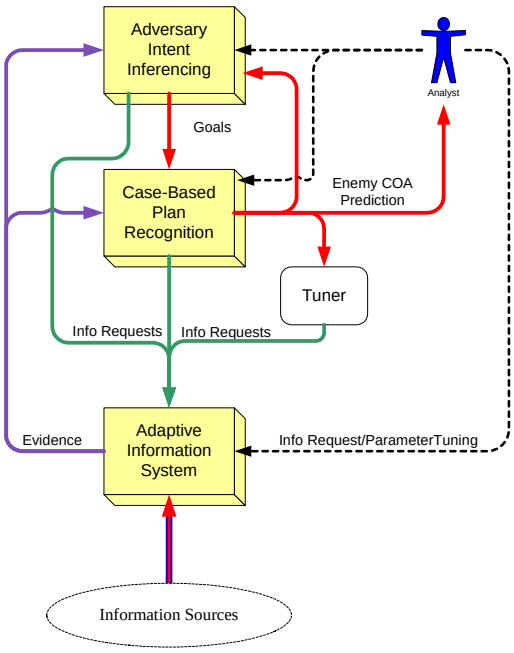


Figure 6. Big Picture

To summarize, the functionality of the hybrid case-based plan recognition system is as follows.

1. To produce predictions of enemy COAs based on predicted red and known blue intentions.
2. To anticipate possible interactions of red and blue given current information by requesting states to be monitored by AIS. The anticipations may be both positive and negative. The former signals hypothesized opportunities, whereas the negative signals warnings.
3. To update predictions and information requests as new evidence arrives and as inferred red or blue goals change.
4. To produce warnings to analysts when actions on one blue component interferes with the commitments of another blue component.

Adaptive Information System (AIS). The AIS is responsible for providing information from various sources to provide evidence for the other two components. The information provided is based on analyst input, requests for further information from the Plan Recognizer, AII, or new information requests based on predicted enemy courses of action. Requests for further evidence may be requests for current or past evidence, or they may be requests for future information as it becomes available. The AIS is tunable based on time or quality. The AIS can be asked to provide the best information possible by a certain deadline or to provide information at a specified confidence level. The AIS consists of an organization of intelligent information agents that reorganize to provide the required information while reducing communication and processing overhead. The individual information agents are relatively simple and only know how to produce certain pieces of information. This information can be gathered directly from known information sources or by combining outputs from known sources or other information agents. The AIS adapts to changes in information requests by modifying its organization to provide the right information, at the right time, at the right quality. Adding new intelligent information agents and updating the team's organizational knowledge [11] can extend the AIS incrementally during execution. This capability allows the AIS to automatically adapt quickly to changing

requirements as well as to long term changes in the information environment without having to "rebuild" the entire system.

Tuner. The fourth component shown Figure 6 is the AIS "Tuner", which is capable of automatically modifying the AIS information requests based on the enemy courses of action predicted by the Plan Recognizer. This component is not integral to the overall system and may not be included in the initial versions of the system.

Summary. We have presented a unified approach to incorporating the adversary into an online system for explanation and prediction of adversary behaviours and actions. It takes into account adversary reactions to blue force actions and addresses the need to retask information gathering resources to better tune adversary predictions. We presented an architecture that can be integrated into blue forces mission planning and execution.

PROJECT STATUS AND RELATED EFFORTS

In this paper, we have presented a cognitive architecture for adversarial intent inferencing for use in future mission planning systems. The Adversarial Intent Inference for Predictive Battlespace Awareness Project is a basic 6.1/6.2 research project sponsored by the AFRL/IF's Information Institute Research Program. Our goal has been to design and develop advanced adversarial modeling and prediction tools that can provide the necessary enabling technologies to support AF mission planning and execution as well as wargaming needs such as EBO, Predictive Battlespace Awareness (PBA), Intelligent Preparation of the Battlespace (IPB), Information Fusion, etc. Currently, we have built a prototype system to model the adversary's behavior based on probabilistic models (Bayesian Networks/Influence Diagrams) and evaluated its effectiveness as a proof-of-concept. Our prototype simulates the "Battle at Khafji" scenario during the Persian Gulf War by predicting and updating the predicted actions of the adversary over time as the events unfolded.

The architecture has also been inserted into military wargaming environments [29]. In the current world environment, the rapidly changing dynamics of organizational adversaries are increasing the difficulty for Military analysts and planners to accurately predict potential actions. As an integral part of the planning process we need to assess our planning strategies against the range of potential adversarial actions. This research project investigates the feasibility of utilizing AII as a core element within a predictive simulation to establish emergent adversarial behavior. It is our desire to use this intelligent adversary to *generate alternative futures* in performing Course Of Action (COA) analysis. Such a system will allow planners to gauge and evaluate the effectiveness of alternative plans under varying actions and reactions.

ACKNOWLEDGEMENTS

The author would like to acknowledge the significant contributions to this project made by Sergio Gigli and Frank Vetesi (Lockheed Martin, Advanced Technology Laboratories), Robert Hillman (Air Force Research Laboratory, Information Directorate), Joshua Surman (University of Buffalo), and Ben Bell (CHI Systems). I would also like to especially thank John Graniero (Air Force Research Laboratory, Information Institute), Don Monk (Air Force Research Laboratory, Human Effectiveness), and Scott Brown (USAF) for their tremendous support towards establishing this research project. This work has been supported in part by a grant from the Air Force Research Labs, Information Directorate, Grant No.F30602-01-1-0595 through the Information Institute Research Initiative.

4.5 Multiple Strategy Generation for War Gaming

Timothy Revello[17], Robert McCartney[28], Eugene Santos[39]
Computer Science and Engineering Dept., Univ. of Connecticut,
371 Fairfield Rd., U-1155, Storrs, CT 06269-3155

ABSTRACT

In this paper we present a framework for the automated generation of strategies that accounts for the multiple kinds of uncertainty found in war games, provides for a domain independent approach to strategy generation, and results in robust strategies. Our approach is to sample over multiple trials for varying victory conditions, different threat profiles, and variable system performance to achieve a degree of independence in the resulting strategy. This allows a search for robust strategies versus those that are effective only under specific conditions. War games have uncertainty in what is needed to achieve victory, in system performance, and in threat behavior. There are multiple options for forces, employment, and warfare styles. All these factors combine to produce a large, complex space of possible solutions or strategies. Through the use of powerful search techniques like evolutionary computation and modern computing assets it has become practical to search this space for strategies with robust performance. Our framework is modular in nature, allowing a variety of search techniques, warfare scenarios, system models, and other parameters to be interchanged. In the paper the framework described above is demonstrated using an antisubmarine warfare scenario. Evolutionary programming techniques are used to search the space of possible strategies.

Keywords: war gaming, strategy generation, multiple strategies, evaluation, evolutionary computing, evolutionary programming, adversary modeling

1. INTRODUCTION

War gaming is a technique that is currently in wide use in both military and nonmilitary organizations as a decision making and training tool. War games are models of real world conflicts with the degree of fidelity varying with game goal. They have uncertainty in what is needed to achieve victory, in system performance, and in threat behavior. There are multiple options for forces, employment, and warfare styles. All these factors combine to produce a large, complex space of possible solutions or strategies. Consequently there are numerous challenges in representing and playing war games. The focus of this paper is the presentation of a framework for the automated generation of strategies that addresses current challenges in war gaming. The framework we present accounts for the multiple kinds of uncertainty found in war games, provides for a domain independent approach to strategy generation, and results in robust strategies. It is modular in nature, allowing a variety of search techniques, warfare scenarios, system models, and other parameters to be interchanged. Using the framework and sampling over multiple trials for varying victory conditions, different threat profiles, and variable system performance, a degree of independence in the resulting strategy can be achieved.

War games are models of real world conflicts and are characterized by uncertainty and complexity. There are two types of uncertainty. The first is uncertainty associated with variables

[7] revellote@npt.nuwc.navy.mil; phone 401 832-8256
[8] robert@engr.uconn.edu; phone 860 486-5232
[9] eugene@cse.ucon.edu; phone 860 486-1458

working inside of the framework of the game rules. Opponent's moves, dice rolls, and the location of pieces all have uncertainty associated with them. Most games such as checkers and backgammon contain this type of uncertainty. The second type of uncertainty that is found in war games but not in traditional games is uncertainty in the game rules themselves. In a war game, as in real conflicts, the specifics of what is required to win or cause the opponent to concede is generally not known in advance. As an example, it is not known in advance how much punishment a military opponent will accept before surrendering. They may be willing to fight to the end. They may be only willing to tolerate a certain level of economic and military damage before conceding the battle in hopes of ensuring regime survival. They may be more sensitive to damage in one area than another. There may be more than one way to win. It may be possible to win by defeating opposing military forces, by imposing economic isolation, by removing key members of the senior leadership, or by some combination of these factors.

Uncertainty in the rules is an important characteristic to capture since it impacts the effectiveness of any strategy used.

War games can have one to many active players. The game pieces or forces involved are typically numerous. Interrelationships between the pieces tend to be complex. A force may be capable of engaging a variety of types of opposing units but have a different effectiveness against each of them. The location of pieces may be unknown and there is typically uncertainty associated with their effectiveness. The playing positions in a war game are not mirrors of each other as they are in traditional games like chess or checkers. War games are typically tailored to the scenario being gamed. Each game is usually custom designed to fit some scenario of interest. Unlike traditional games, there are no standard war games that are replayed in the exact same form repeatedly.

Some issues and challenges in contemporary war gaming are discussed in [Bracken and Shubik, 2001; Rubel, 2001; McCrabb and Caroli, 2002]. One of the areas of primary concern is that current war gaming techniques are not sufficient for exploring new styles of warfare. There is a need to be able to depict the effect of a variety of events on decision makers. There is a need to include uncertainty which is an overriding factor in actual conflicts, but is minimized in many war games. This is partly because of a lack of adequate techniques for modeling and analyzing its impact. Included in uncertainty is the need to better account for not knowing what will trigger events such as attacks, withdrawals, and surrenders. Another challenge is to develop automated techniques that allow the exploration of as many courses of action as possible. Manual play limits the number of games as well as concepts and courses of action that can be addressed. Because of time and resource constraints, war games tend to address a specific scenario. The resulting strategy tends to be tailored to that scenario. It may not be robust as it does not take into account possible alternative opponent strategies and their implications. There is a need to address complex multiplayer games that contain noise and uncertainty. Games must include not only attrition warfare but effects based, network centric, and other additional styles of warfare.

Automation is cited as a way in which to address some of the challenges in contemporary war gaming. Computing technology has been used in war gaming since the introduction of mainframe computers. Initially it was used for managing and displaying data. Later, artificial intelligence techniques were used for problem solving and generating automated opponents [Perla, 1990]. While there are a number of pitfalls associated with automated gaming and opponents, it is recognized that there can be many benefits. Automation offers the opportunity to more completely explore potential solutions and strategies. Automating adversary behavior and intent inferencing is discussed in [Surman, Hillman, and Santos, 2003; Santos, 2003]. Evolutionary computation (EC) has been discussed as a method for potentially generating

strategy and tactics in these complex environments which involve multiple factors and uncertainty [Ilachinski, 1996]. Recently, EC techniques have begun to be applied to war gaming. Much of this effort has been in the area of agent based simulations using simple agents defined by a small set of behavior rules and attributes. Here combat is viewed as emergent, arising from the collective interactions of the combatants. The focus is on discovering the patterns of group behavior that generates the desired overall result. Some efforts to study emergent behavior and its relationship to combat are described in [Ilachinski, 2000; Taylor, 2001; Gill and Grieger, 2003; Hill, McIntyre, Tighe, and Bullock, 2003].

The framework we describe addresses these war gaming challenges. An automated, modular approach is taken. The core of the approach is a simulation and rule set that define the war game to be played. Around this is added an evolutionary computation search technique. A population of strategies is generated and each strategy is scored using the war game. The evolutionary search process generates offspring and selects successive generations. This process continues until a halting criterion is satisfied. The modular approach allows a variety of warfare scenarios, evolutionary computation search techniques, and other models to be interchanged. During the scoring of strategies, multiple trials are conducted in order to generate a score. Threat strategies, game rules, and system performance are varied during the trials. This approach allows a search for robust strategies or solutions that are not tailored to a very specific scenario. Automation enables the exploration of large solution spaces. Instead of addressing a very specific scenario with a specific list of assets, lists of choices for concepts and courses of action can be chosen from. The implications for uncertainty in opponent behavior can be explored. In the following the framework is described in detail. This is followed by a description of an experiment that demonstrates the use of the framework. This proof of concept experiment explores an antisubmarine warfare (ASW) scenario. Evolutionary programming (EP) techniques are used to search the space of possible strategies. Both effectiveness and force cost are accounted for in the objective function.

2. FRAMEWORK DESCRIPTION

2.1 Framework

The framework for the automated generation of war game strategies we propose addresses current challenges in war gaming including the need to address uncertainty, the effects of events on decision makers, exploring many courses of action, and multiple warfare styles. In describing the framework, a game involving two sides is considered. The side for which a strategy is being searched for or developed is referred to as the Blue side. The opponent is referred to as the Red side. In addressing the uncertainty in the game rules, multiple criteria for victory for each side are used. Criteria thresholds are varied from trial to trial. They are drawn probabilistically from within a defined range. A range of strategy options are defined for the opponent and the strategy used varies from trial to trial. This addresses the uncertainty in opponent behavior. Using game rules and constraints, the effect of events on decision makers can be represented. Outcomes of events can cause shifts in opponent behavior profiles. An opponent might become more conservative or more aggressive. Losses might cause certain assets to be withdrawn and conserved but not trigger surrender. In constructing Blue strategies, there are options for assets or resources that can be applied. These can be employed at a variety of times and locations, the combination of which defines the style of warfare. The possible combinations of resources, time of employment, and place of employment define the size of the solution space of Blue strategies. Given this space, a search methodology using EC techniques is applied to find strategies that perform well.

In order to define a modular framework and search methodology for war games, it is necessary to break the game into its component parts. This is shown in Figure 1. In war games, as in all games, there are rules that define allowed actions and conditions for winning. This specifies what is required to win for each side. There may be more than one way of winning for a given side. Criteria associated with each rule are used to keep track of progress toward the goal of winning. Associated with these criteria is uncertainty. Unlike traditional games such as chess or checkers for which the rules are well defined, the rules for war games are uncertain. It is typically unclear exactly what is required to win though a general sense of what is required may exist. An opponent might withdraw when 25 percent of his forces have been lost or it might require 40 percent losses. In some cases all that is required to win is to not lose, or to prevent the opponent from meeting his criteria for winning. There may be general criteria for winning such as destroying a given percentage of the opposing force. This is a typical attrition warfare approach. Special criteria for winning can also be defined. As an example, the opponent may withdraw from a battle if his command and control capability is destroyed. This is an example of effects based warfare. By destroying command and control the opposing forces have been neutralized without physically destroying them.

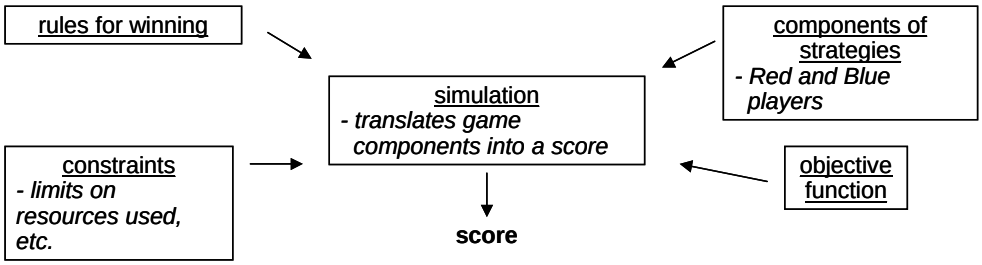


Figure 1: War Game Components

Constraints are often used in war gaming. There are typically limits on the amount of available resources. There are usually limits on quantities like when platforms are available, how many days may be spent at sea, and weapon load outs. Constraints can also apply to what target sets can be addressed at various times and other conditions for conducting attacks. Through the use of constraints, a variety of rules of engagement (ROE) can be implemented.

Each player in the game will have a strategy. A strategy is a plan of action to accomplish a specific goal. The strategy can be thought of as being composed of the resources used, when they are applied, and where they are applied. There can be more than one way to accomplish the goal of the strategy. Each may have a different criteria for winning associated with it. The strategy may be composed of one or more substrategies, each of which pursue a different criteria for victory. These substrategies may be pursued in parallel, in serial, or some combination of the two.

The war game will also contain a simulation of the type of warfare being conducted. Before the application of computers to war gaming, the movement of pieces on the game board was used to simulate warfare. In the current case, computer simulations of warfare to varying degrees of fidelity will be used. Automation is important since the goal is to search very large spaces of possible strategies for those that perform well. A large number of games must be played in order to accomplish this. The simulation is used in conjunction with an objective function in order to obtain a score for a particular strategy. They translate the components of the strategies of the opponents into a score. The objective function can emphasize various quantities. These quantities may reflect considerations relative to higher or national level strategies or policy for the

various sides. Winning is of course the primary goal but there may be other important considerations. These include winning quickly, minimizing own force losses, minimizing the cost of the force used, or minimizing collateral damage. For a set of strategies for the Red player, there is a set of strategies for the Blue player that result in winning the game. Either of these sets can range from being the null set to having one or more set members. The objective function determines the relative values of the Blue strategies in the set of winning strategies.

Having defined the component parts of the war game, they can be fitted into a framework to which evolutionary search techniques can be applied. Figure 2 shows the framework for use in the automated generation of strategies. The approach taken is modular in nature. The simulation could be a variety of possible models. It could be land or sea combat. It could be a high or low resolution model. It could be a game at the operational or strategic level. Depending on the type of simulation used, appropriate models can be linked to the simulation. These can describe search, the way combat is conducted, how weapon systems perform, or similar functions. Different models can be interchanged for various experiments.

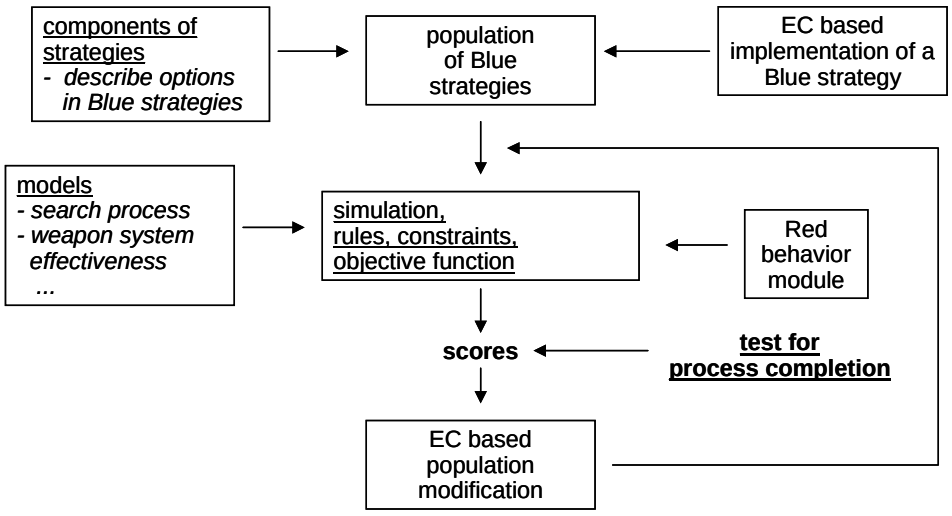


Figure 2: Framework

In Figure 2 the goal is to search for Blue strategies that perform well versus the Red opponent. The Red behavior module describes how Red will behave. This includes uncertainty over what strategy Red will employ. There may be several possible courses of action for Red. They might be chosen probabilistically or selection may be event driven. It is possible to interchange different modules from different sources in conducting various experiments due to the modular nature of the framework.

An evolutionary computation based search process is used to find Blue strategies with good performance. There is an initial population of Blue strategies. The initial population is typically generated at random. Individuals in the population are evaluated using the war game and simulation to generate scores. The population is then modified using an EC technique to produce a new generation of individuals. The process is repeated until some halting criterion is met. There are a number of possible EC search techniques that could be used. This is again part of the modular nature of the framework. Two example methods are genetic algorithms and evolutionary programming. EC based methods work by modifying the population to produce offspring. A

new generation is composed using the existing individuals and offspring based on fitness and probability. The application of EP to the framework is demonstrated in the example experiment which follows.

2.2 Problem Types That Can Be Addressed

In Figure 2, the configuration shown is for searching for Blue strategies given a behavior module that describes possible Red behaviors. A number of types of problems can be addressed in this configuration. The first is an analysis of Blue alternatives. Strategy components for Blue are populated with different technology and platform options. Included is where and when they can be employed. The search addresses combinations of force mix and employment method that perform well against the range of possible Red behaviors. This approach can be applied to concept analysis and planning types activities [Davis, 2002]. Another type of problem that can be addressed in this configuration is that of concepts of operation. The force mix for Blue can be specified and the search can address the best method of employing the forces. In either of the two cases above, the issue of rules of engagement can be explored. ROE are modeled through the use of various constraints. The impact of ROE changes on Blue alternatives or concepts of operation can be studied. The issue of uncertainty in ROE can also be addressed. The combatant commander often does not know how ROE will change or when. It may be beneficial to study the robustness of strategies in the face of uncertain ROE.

The process in Figure 2 could also have been structured a different way to answer a different type of question. The Red and Blue positions can be interchanged. Given a Blue strategy there may be a desire to test it against an opponent to see how robust it is. In this case a list of Red options or a fixed Red order of battle is used. A search is then conducted for Red strategies that could be effective against the Blue strategy being tested. This allows the exploration of variations of known Red strategies as well as the search for previously unknown Red strategies that might be effective.

If vulnerabilities are found to the Blue strategy, the entire process can be iterated. Effective Red strategies can be added to the Red behavior module and Blue strategies that have good performance can then be searched for. The Blue strategy can then be held fixed and the space of Red strategies explored to search for Blue vulnerabilities. This process can be repeated until Blue strategies of sufficient robustness are found or issues preventing Blue from achieving its goals are identified. The phased evolutionary process described differs from coevolution in that in phased evolution only one side at a time changes while in coevolution both sides change simultaneously. This allows a thorough exploration of the solution space associated with each iteration. The solution space changes between iterations, with the amount of solution space overlap between iterations being dependent on the specifics of the problem.

3. EXAMPLE EXPERIMENT

3.1 Overview

In this section the proposed framework for the automated generation of war game strategies is demonstrated in an example experiment. An operational level antisubmarine warfare scenario involving two sides is the subject of the game. While composed of simple rules and parts, the interaction of the numerous rules and components produces a large, complex game space. The modular framework components used are described in detail. This is followed by a discussion of the results of the experiment which determines strategies consisting of force mix and employment. The side for which a strategy is being determined is referred to as the Blue side.

The opponent is referred to as the Red side. Red behavior is composed of a number of possible Red strategies or courses of action against which the Blue solutions are tested.

3.2 Framework Components

The components of the war game used in the experiment are discussed in this section. They correspond to the framework components shown in Figures 1 and 2. The rules, constraints, objective function, and simulation used are described first. Figure 3 shows the game board used in the antisubmarine warfare game. Blue's goal in the game is to operate a carrier battle group (CVBG) in area 11 for five weeks. Red's goal is to prevent Blue from achieving its goal. There are 11 operating areas on the board. Areas 1 through 4 are notionally 2500 square nautical miles (SQNMs) each while the remaining areas are 22,500 SQNMs each. The Red force is composed only of submarines. There are two types, diesel submarines (SSs) and air independent propulsion submarines (AIPs). The Blue force has nuclear submarines and three types of surface ships as options. Red submarines enter the board from their bases in areas 1 or 2 and may move to any area. Blue ships enter the board in area 10 and can move to any area except area 1. It is assumed that Blue will not move past areas 3 and 4 into the straits due to increased risk. Red submarines in area 1 can move to area 3 or 4. Surface ships and submarines in areas 2 through 11 can move to any adjacent area. Surface ships can move two areas in a day. Submarines, which are assumed to be operating slower than surface ships, can move one area per day.

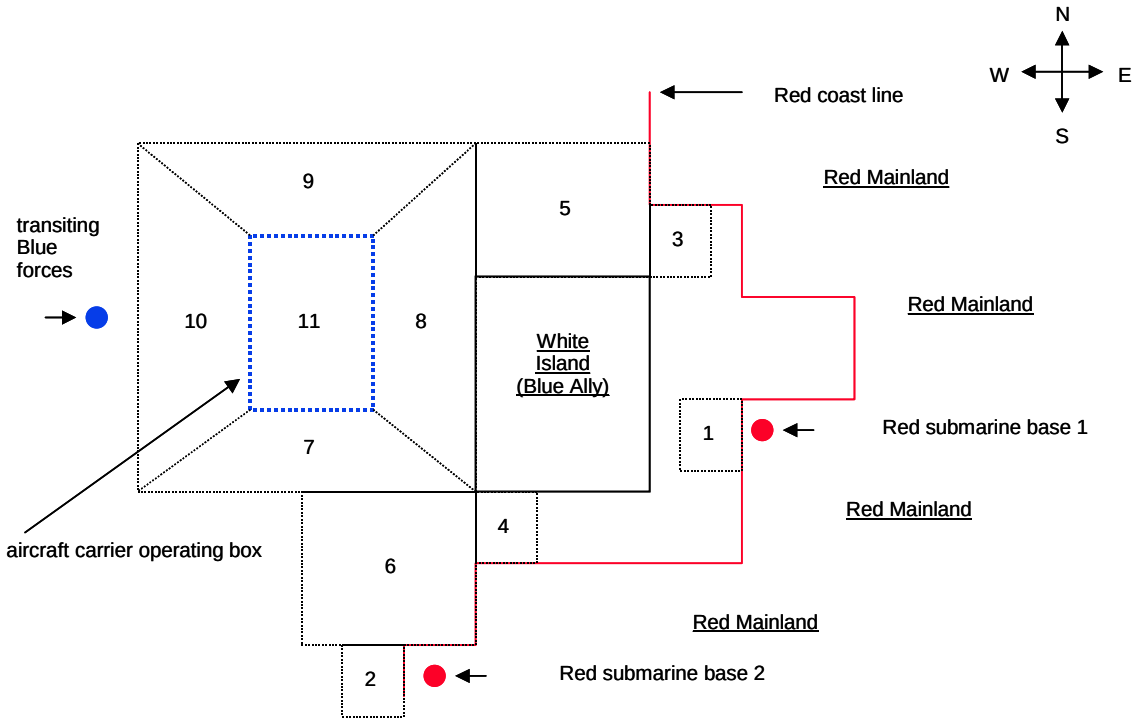


Figure 3: Game Board

There are several ways of winning the game for each side. These criteria account for the uncertainty in the conditions needed to achieve victory. Blue attempts to transit a CVBG containing three aircraft carriers (CVs) into area 11 and operate it for five weeks. Blue can win by achieving this goal or by causing Red to withdraw its submarines. If Red loses between 25 and 35 percent of its total submarine force it will withdraw from the conflict to preserve the remainder of the force. The percentage used is chosen probabilistically from a uniform

distribution for each game or trial played. In addition, if Red loses 25 to 35 percent of its AIPs, the remainder of the AIPs will return to port. Red will continue the conflict using its remaining SSs. This percentage is again chosen randomly from a uniform distribution for each trial. AIPs are more capable and more expensive than SSs and subject to additional conservation measures. Blue can lose in two ways. If one or two of the three Blue CVs is sunk Blue loses. This number is chosen randomly with 0.5 probability for both outcomes. Also, if Blue loses either ten escort ships or escort ships with a total value of 6.0 then Blue loses the game.

Blue's goal is to win but there is an additional objective. Blue should win but also minimize the cost of the force used to achieve victory. It is not expected that the force can be optimized since there are a number of sources of uncertainty in the game. However, the force should be sized appropriately to win given uncertainty. It is assumed that Blue will accept some risk since sizing a force to ensure victory under any conditions can often be prohibitively expensive. The objective function used for Blue is the following. The score is the percentage of the trials in which Blue wins minus a cost factor. The percentage of trials portion of the score has a maximum value of 0.95, with Blue accepting a 5 percent risk factor. The cost factor is the cost of the Blue force used divided by 200. Previous efforts have shown that including cost factors that are small in relation to benefit factors can reduce complexity of the solution and increase performance [Revello and McCartney, 2002].

Figure 4 shows an overview of the simulation used in the experiment. Both Red and Blue forces are initialized. The game then runs for a maximum of six weeks. The game is broken into 24 hour units. During a 24 hour unit, Blue surface ships, Blue submarines, and Red submarines each take their turns. During a turn, movement, search, and attack are conducted. A ship can only move to the next adjacent area in a single turn. Blue surface ships receive an additional turn at the end of the 24 hour unit since they are modeled as having a speed of advance twice that of submarines. At the end of the 24 hour unit a check is made to see whether orders are changed for any of the ships. The orders dictate the ship's destination and behavior. Ship's orders are affected by event driven variables which change during game play. Blue's CVBG enters the game on day 8. This allows for actions by both Blue and Red prior to CVBG arrival. The game runs until a victory criteria is met or the game ends after six weeks.

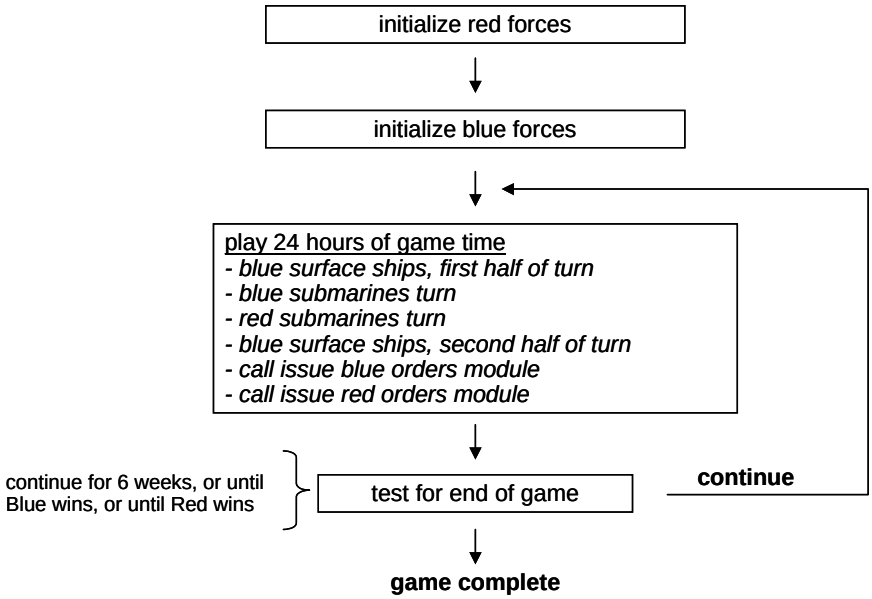


Figure 4: Overview Of Simulation

The models used in the game are part of the modular framework. In this game each of the four Blue ship types have specified search rates versus the two types of Red submarines. Blue ships conduct searches and attacks only versus Red submarines in their areas. Probability of detection for Blue searching for Red is computed using the area search equation [Naval Operations Analysis, 1977]. Length of search is 12 hours for surface ships and 24 hours for submarines. Probability of kill and counterkill for Blue attacks varies by searcher and target. In this game it is assumed that a Red submarine can detect all Blue surface ships in its area. It is assumed that Red has no search capability versus Blue submarines. Probability of kill and counterkill for Red attacks also varies by searcher and target. Search rates and weapon system effectiveness are input as tables. Blue ships are organized into groups with each group containing only one type of ship and operating independently. Red submarines always operate in groups of one. Multiple groups from either side can be located in the same area. Each Red submarine and Blue ship group can make one attack per turn. Each Red submarine randomly selects a Blue group operating in its area to attack. If a kill is achieved the number of ships in the Blue group is decremented by one. A Blue group also randomly selects a Red submarine in its area for search and attacks. Each Blue ship in the group attacks the same Red target.

Another component of the framework is the Blue and Red strategy components. The Red force contains 14 diesel submarines and 8 air independent propulsion submarines. The former carry 16 torpedoes and the later carry 8 torpedoes and 8 antiship cruise missiles (ASCMs). The mix of ships in the two Red submarine bases, when they depart their homeport, and their destinations varies with Red behavior and will be described later. Blue's ASW strategy consists of which type of assets to use, where to use them, when to use them, and their patrol patterns. This combination of variables is the subject of the EC based search. There are four types of ships available to Blue: nuclear powered submarines (SSNs), frigates (FFGs), destroyers (DDGs), and small ASW escort carriers (CVEs). The relative costs of the four types in order are 2.5, 0.2, 1.0, and 0.4. Blue ships have four options for patrol behavior. In the first, the ship alternates between searching area 11 and the areas adjacent to area 11. The second is to transit to a specified area and remain there on station. The third is to conduct a circular patrol around area 11 through areas 7 to 10. The fourth

is a random patrol in which the ship moves in a random direction, searches the area, then moves in another random direction.

Different Red behavior modules can be plugged into the strategy generation framework. In the experiments performed a Red behavior module describing a composite behavior was used. Blue has postulated four primary Red modes of behavior. Red submarines may operate independently or in loose knit packs with submarines moving from area to area in unison. Red submarines may be in a conservative posture or an aggressive one. In a conservative posture Red does not enter area 11 and does not attack the Blue CVs. No Red attacks are conducted until after the Blue CVs arrive on day eight. In the aggressive posture Red will attack any Blue ship in any area starting on day one. The combinations of independent/pack and conservative/aggressive yields four possible combinations or modes. Since Blue is uncertain of Red behavior, a mode is randomly chosen for a given trial with all modes being equally likely.

Red submarines have three basic patrol patterns which are shown in Figure 5. The first pattern is to deploy to a given area, remaining there attacking targets of opportunity when authorized. At least some Red submarines patrol in this pattern in all four of the behavior modes. The second pattern is to patrol in a circle around area 11 attacking targets of opportunity when authorized. This is used only in the pack conservative combination. The third pattern is to move into area 11, remain for one day conducting attacks, move to a random adjacent area for one day, and repeat the sequence. This is used in both aggressive modes. The number of submarines in the different patrol patterns varies with mode. In the independent conservative mode, all submarines use the first patrol pattern. In the aggressive modes only a few use the first patrol pattern, protecting the submarine transit lanes to their bases against Blue ships.

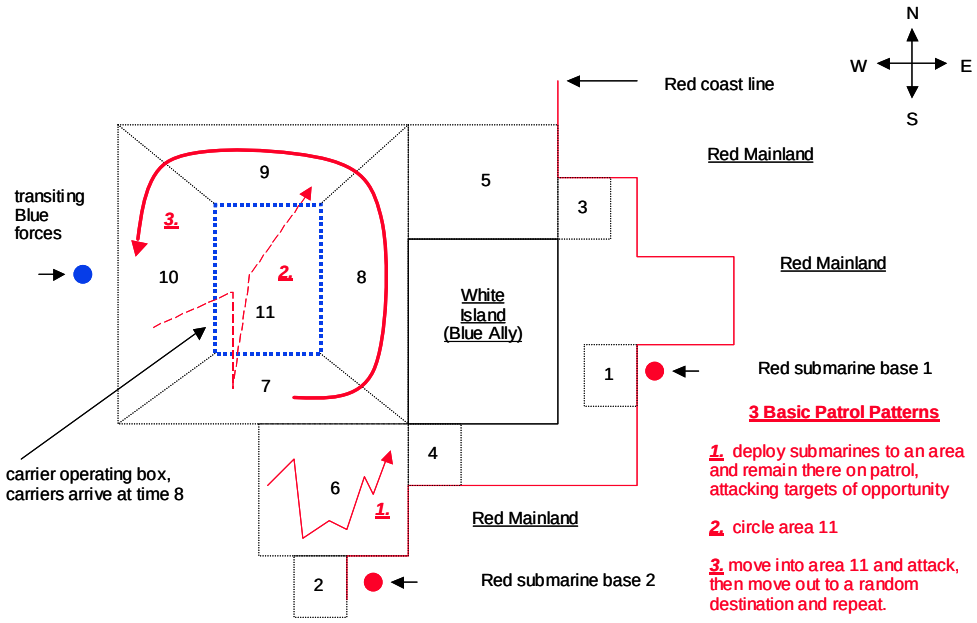


Figure 5: Red Patrol Patterns

There are additional rules that govern individual Red submarine behavior. Each submarine can remain at sea for three weeks. After three weeks the submarine must return to port to replenish supplies. During this time weapons are also reloaded to the maximum carried. This takes two weeks, after which the submarine can return to sea. Each submarine has a fixed weapon load out.

Weapons are always fired in salvos of two. When there are only two torpedoes remaining, the submarine returns to port to reload. In the conservative mode, Red submarines will not attack until after the CVs arrive on day eight. If Blue should initiate hostilities prior to this time, Red will shift from a conservative to an aggressive profile. Red submarines will be given new destinations and patrol patterns.

The EC search process that is applied to the game is depicted in Figure 6. A population of 30 Blue strategies is used. Each Blue strategy is composed of 20 ship groups or agents. Nineteen of these are variable with the twentieth being composed of the three CVs. Each of the 19 variable ship agents can contain from zero to nine ships all of the same type. There are four possible ship types. Each of the variable Blue agents has one of the four possible behaviors described earlier. The first two behaviors have an initial destination that can be selected from areas 2 through 11. The initial destination for behavior 3 is area 10 and the initial destination for behavior 4 is area 11. Each variable agent can enter the game on day 1, 8, 15, 22, 29, or 36. Agent 20, which contains only the three CVs, enters the game on day eight, transits to area 11, and remains there. Given the composition options for each variable agent and a population with 19 variable agents, the Blue solution space is on the order of 10 to the 70th power.

EP is the EC search algorithm used in the experiment. Thirty offspring are generated using a Gaussian distribution and 30 are generated using a Cauchy distribution. Since the vector elements or attributes of the variable agents are discrete values, Gaussian and Cauchy like distributions are used during the mutation operations. Gaussian mutations tend to be shorter distances from the current value than are Cauchy distribution distances. During reproduction, each of the 90 individuals is compared to 10 randomly selected individuals. The 30 with the best score become the new generation [Yao, Liu, and Lin, 1999].

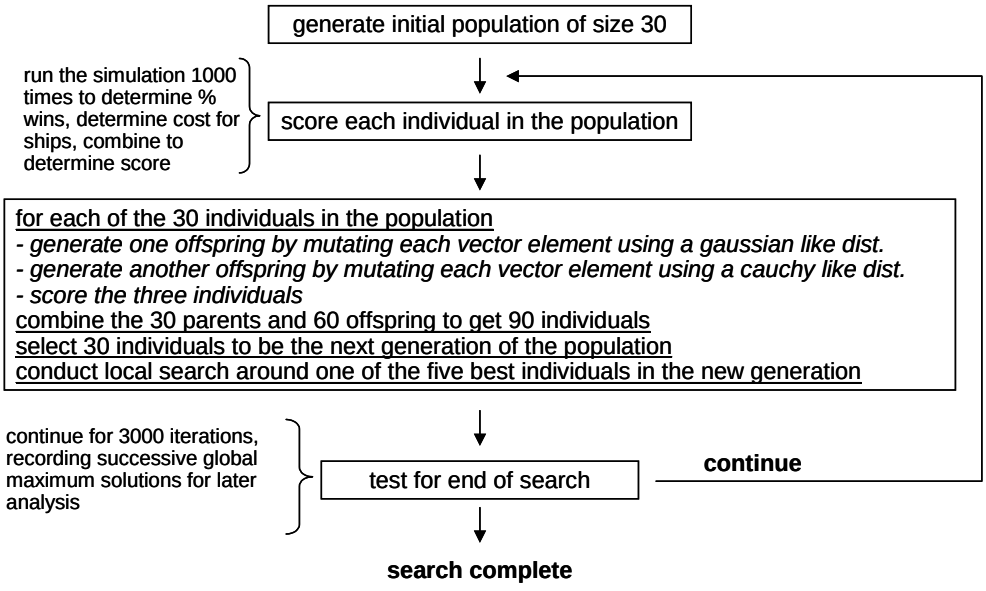


Figure 6: EC Search Process

3.3 Experiment Results

Through the use of the framework components, we were able to address the war gaming challenges enumerated earlier in the paper. In the example game, uncertainty, the effects of events, rules of engagement, courses of action, and elements of different warfare styles have been

represented. In this context a search for Blue strategies that perform well was conducted using EP. Three thousand generations consisting of 30 individuals each were run. Each Blue individual is a strategy for playing the ASW war game. Blue strategies that were successful in achieving the Blue goal given the game uncertainties were found. The scores for the strategies varied with Blue resources used. A global maximum strategy is defined as a Blue strategy with the best score or performance to date. During the experiment, each strategy that constituted a global maximum was saved for later analysis. Figure 7 shows a plot of the number of ships in the global maximum as a function of generation. As time increases, the number of ships in the Blue strategy and the resultant complexity decreases. The best performing individual found was in generation 323. This strategy contained two ship groups in addition to the group containing the three CVs. One group contained three FFGs and the other group contained four CVEs for a total of seven ASW escort ships used. Both groups enter the game on day one and transit to area 11, remaining there on station. The two groups work to clear area 11 of any Red submarines prior to the CVs arrival and serve as escorts after the CVs arrive.

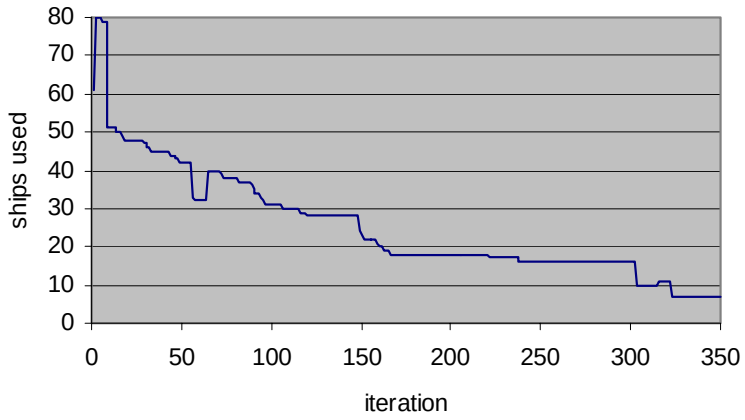


Figure 7: : Blue Ships In The Global Maximum

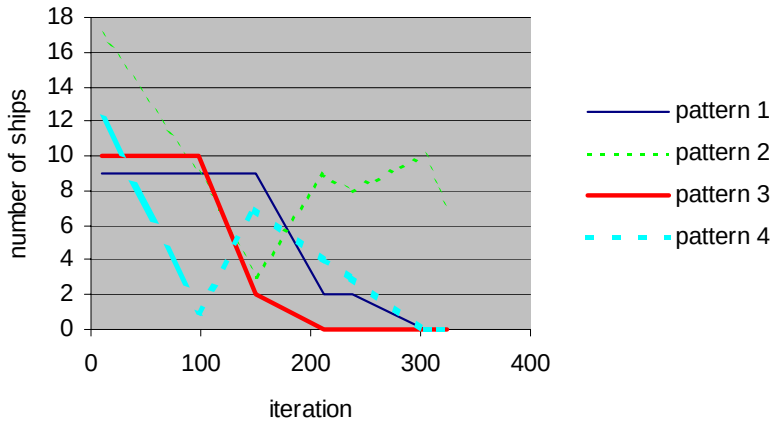


Figure 8: Blue Patrol Types Conducted

Blue ships have four options for patrol behavior. The first is to alternate between searching area 11 and the areas adjacent to area 11. The second is to transit to a specified area and remain there on station. The third is to conduct a circular patrol around area 11. The fourth is to move in a random direction, search the area, then move in another random direction. In Figure 8 the number of ships conducting each of the four types of patrols as a function of time is shown for selected global maximums. As can be seen, a variety of strategies are tested during the search for Blue strategies that perform well. In the early stages, there are ships located in area 11 defending the CVs as well as ships outside of area 11 conducting various types of patrols. As time increases, the ships patrolling outside of area 11 are reduced and eventually eliminated. The inclusion of the cost term in the objective function exerts pressure on the evolutionary search to find efficient solutions. The seven ASW escorts operating in area 11 are sufficient to protect the CVBG given the four postulated modes of Red behavior.

4. SUMMARY

In this paper we have presented a framework for the automated generation of strategies that accounts for the multiple kinds of uncertainty found in war games, provides for a domain independent approach to strategy generation, and results in robust strategies. The framework is modular in nature, allowing a variety of search techniques, warfare scenarios, system models, behavior modules, and other parameters to be interchanged. It can be applied to a variety of problems including concept analysis and planning types activities and analysis of threat courses of action. Using the framework and sampling over multiple trials for varying victory conditions, different threat profiles, and variable system performance, a degree of independence in the resulting strategy can be achieved.

Through the use of the framework components, we were able to address current war gaming challenges in an example ASW war gaming experiment. Uncertainty, the effects of events, rules of engagement, courses of action, and elements of different warfare styles were represented in the experiment design. The use of the framework in the automated search for strategies given game and opponent behavior uncertainties was demonstrated. EP was used to search the space of possible Blue strategies. Blue strategies that were successful in achieving the Blue goal were found. Scores for Blue strategies varied with resources used. The cost term in the objective function exerted pressure to reduce the resources used, resulting in a reduction in solution cost and complexity over time. Ongoing work in this area includes an effort to implement phased evolution in order to test Blue ASW strategies by searching for previously unknown Red strategies that represent Blue vulnerabilities.

5. ACKNOWLEDGEMENTS

This work was supported in part by Air Force Research Labs, Information Directorate, Grant No. F30602-01-1-0595. Special thanks to Robert Hillman for his insights and encouragements to this effort.

6. REFERENCES

Bracken, P., Shubik, M. (2001). "War Gaming In The Information Age", Naval War College Review, Vol. LIV, No. 2, Spring 2001.

Davis, P. (2002). Analytic Architecture for Capabilities-Based Planning, Mission-System Analysis, and Transformation, RAND, Santa Monica, CA.

Gill, A. and Grieger, D. (2003). "Comparison of Agent Based Distillation Movement Algorithms", Military Operations Research, Vol. 8, No. 3, 2003.

Hill, R., McIntyre, G., Tighe, T. and Bullock, R. (2003). "Some Experiments With Agent-Based Combat Models", Military Operations Research, Vol. 8, No. 3, 2003.

Ilachinski, A. (1996). Land Warfare And Complexity Part II: An Assessment Of The Applicability Of Nonlinear Dynamics And Complex Systems Theory To The Study Of Land Warfare, CRM 96-68, Center For Naval Analysis, Alexandria, VA.

Ilachinski, A. (2000). "Irreducible Semi-Autonomous Adaptive Combat (ISAAC): An Artificial Life Approach To Land Combat", Military Operations Research, Vol. 5, No. 3, 2000.

McCrabb, M. and Caroli, J. (2002). "Behavioral Modeling And War Gaming For Effects-Based Operations", Proceedings Of The Analyzing Effects Based Operations Workshop, Military Operations Research Society, Vienna, VA, January 2002.

Naval Operations Analysis (Second Edition) (1977), p. 127, Naval Institute Press, Annapolis, MD.

Perla, P. (1990). The Art Of Wargaming, United States Naval Institute, Annapolis, MD.

Revello, T. and McCartney, R. (2002). "Generating War Game Strategies Using A Genetic Algorithm", in Proceedings of the 2002 Congress on Evolutionary Computation, IEEE, Piscataway, NJ.

Rubel, R. (2001). "War Gaming Network Centric Warfare", Naval War College Review, Vol. LIV, No. 2, Spring 2001.

Santos, E. (2003). "A Cognitive Architecture for Adversary Intent Inferencing: Knowledge Structure and Computation", Proceedings Of The SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls, SPIE, Bellingham, WA.

Surnam, J., Hillman, R., and Santos, E. (2003). "Adversarial Inferencing for Generating Dynamic Adversary Behavior", Proceedings Of The SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls, SPIE, Bellingham, WA.

Taylor, D. (2001). "USMC Ground Combat Study Overview", from proceedings of the Complexity: An Important Framework For Understanding Warfare symposium, Center For Naval Analysis, Alexandria, VA.

Yao, X., Liu, Y., and Lin, Y. (1999). "Evolutionary Programming Made Faster", IEEE Transactions on Evolutionary Computation, Vol. 3, No. 2, 1999.

4.6 Constructing adversarial models for threat/enemy intent prediction and inferencing

Eugene Santos Jr. and Allesandro Negri
Intelligent Distributed Information Systems Laboratory
University of Connecticut
Storrs, CT 06269-3155
eugene@engr.uconn.edu

ABSTRACT

We examine an adversary model that captures goals, intentions, biases, beliefs, and perceptions based on a dynamic cognitive architecture that evolves over time. The model manages the uncertainty surrounding the adversary using probabilistic networks. In particular, we consider the challenges of constructing such adversaries and provide solutions towards more effective and efficient engineering of such adversaries. We present the AII Template Generator tool which enables the rapid deployment of adversary models as well as on-demand construction of new adversary components.

Keywords: adversarial intent inferencing, decision making, knowledge acquisition, adversarial modeling, model building

Introduction

Modern elements of war gaming and mission planning and execution ultimately require a computational model of dynamic adversary behaviors. This is especially critical to future operations with regards to predictive analysis and predictive battlespace awareness (PBA) which "…involves studying an adversary to understand what he'll do, how he'll do it, what his capacity to inflict harm will be, and the environment in which he is operating — in short, knowing the scene of the crime before the crime is committed" [4]. In particular, determining an adversary's courses of actions (COAs) involves predicting and evaluating their likely goals, objectives, actions, and desired end states. This is in order to provide the identification and prioritization of the full of set of friendly courses of actions to meet the adversarial threat [12].

As noted in [36]: "In the current world environment, the rapidly changing dynamics of organizational adversaries are increasing the difficulty for Military Analysts and Planners to accurately predict potential actions. As an integral part of the planning process, we need to assess our planning strategies against the range of potential adversarial actions. This dynamic world environment has established a necessity to develop tools to assist in establishing hypotheses for future adversary actions." Thus, without an automated or even systematic computational approach to adversarial decision modeling, the ability to accurately capture and predict (let alone even update) adversary intentions is fundamentally limited by the speed and effectiveness of the human decision-maker, hence, presenting a multitude of challenges with respect to the necessary speed and complexity of current and future military operations. For example, in contemporary war gaming, one challenge is to: "[…] develop automated techniques that allow the exploration of as many courses of action as possible. Manual play limits the number of games as well as concepts and courses of action that can be addressed. Because of time and resource constraints, war games tend to address a specific scenario. The resulting strategy tends to be tailored to that scenario. It may not be robust as it does not take into account possible *alternative opponent strategies* and their implications. […] Games must include not only attrition warfare but effects based, network centric, and other additional styles of warfare." [24] (emphasis added).

In this paper, we present our work in adversary intent inferencing sponsored through the auspices of the Air Force Research Laboratories, Information Directorate and the Information Institute. Our adversary model captures the adversary's goals, intentions, biases, beliefs, and perceptions based on a dynamic cognitive architecture that evolves over time and manages the uncertainty surrounding the adversary using probabilistic networks. Our model has been successfully deployed within multiple war gaming environments to explore the effects of dynamic adversarial behaviour and emergent adversary COAs on potential blue COAs [36, 24]. In particular, we examine the challenges of constructing such adversaries and provide solutions towards more effective and efficient engineering of such adversaries. We present the AII Template Generator tool which enables the rapid deployment of adversary models as well as on-demand construction of new adversary components. We conclude this paper with a discussion of future concepts and directions for adversary intent inferencing.

BACKGROUND: TERMS AND CONCEPTS

In this section, we provide some background on terms and concepts we will be using throughout this paper. For additional details and discussion see [5, 6, 12, 27, 30, 32, 33, 34].

Intent Inferencing. Intent inference involves deducing an entity's goals based on observations of that entity's actions [16, 26]. Such deduction involves the construction of one or more behavioral models that have been optimized to the entity's behavior patterns. Data/knowledge representing observations of an entity, the entity's actions, or the entity's environment (collectively called *observables*) are collected and delivered to the model(s), which match the observables against patterns of behavior and derive inferred intent from those patterns. These inferences can then be passed to an application for generation of advice, definition of future information requirements, proactive aiding, or a host of other benefits. For this paper, we are interested in predicting future adversarial actions, explanation of their behaviors, generation of red courses of actions, and "what-if" analyses to determine the effectiveness of blue courses of actions which is all central to effects-based operations (EBO) [14, 19].

Intent inference can be dissected into three informational components [26, 28, 30, 32, 15]: The first, *interests and focus*, captures at a high level the direction of the entity's attention. The second, *actions and preferences*, describes the activities that can be used to carry out the goals that currently hold the entity's attention, with a focus on how the entity tends to carry them out. The third, *knowledge and reasoning*, provides insight into the deeper motivations behind the goals upon which the entity is focused and illuminates connections between goals. In other words, the first component captures what the entity is doing, the second captures how the entity might do it, and the third infers why the entity is doing it.

Applying the principles of modeling the what/how/why of entity intent, we see that our approach naturally integrates into the major themes of EBO. While the field of individual intent inference has historically focused on better improving the human-system interface, we contend that there is a natural isomorphism between our own prior work in the field of user and team intent inference [9, 11, 12, 28, 30, 31, 32] and the domain of adversary intent inference [6, 12]. While the operational world surrounding an intent inference application would be very different, the inner mechanisms of intent inference map directly between domains as we have demonstrated in [24, 26, 36].

While observables in the user intent domain stem from data collected from human use of systems, observables in the adversary intent domain take the form of tactical information derived from

intelligence databases, observations of the tactical environment, and input from online human experts. In place of window events, keystrokes, and mouse movements, our system in the adversary intent domain uses information about adversary location, movements, and activities to drive its inference. In place of computer state, analyses of information queries, and the content of user dialogue with team members, our system bases inferences on facts about the local terrain, regional weather, and the salient political climate.

Likewise, tactical goals will replace computer operational goals in the results of our intent inference. This would result in identification of an adversary force's objectives and, given models of tactical reasoning, could recommend appropriate reactions. Furthermore, this would indicate expected activity by the adversary and explain the reasons behind that activity. Finally, we can produce alerts of attempted subterfuge or uncover missteps on the part of the adversary.

Uncertainty and Bayesian Knowledge-Bases. We must capture the uncertainties inherent in the adversarial model as well as the uncertainties found in the observables. In probabilistic reasoning, random variables (abbreviated, r.v.) are used to represent events and/or objects in the world. By making various instantiations to these r.v.s, we can model the current state of the world probabilistically. Thus, this will involve computing joint probabilities of the given r.v.s. Unfortunately, the task is nearly impossible without additional information concerning relationships between the r.v.s. In the worst case, we would need the probabilities of every instantiation combination which is combinatorially explosive.

On the other hand, consider the chain rule as follows:

$$P(A1, A2, A3, A4, A5) = P(A1 \mid A2, A3, A4, A5)\, P(A2 \mid A3, A4, A5)\, P(A3 \mid A4, A5)\, P(A4 \mid A5)\, P(A5).$$

Bayesian networks [21] take this process further by making the important observation that certain r.v. pairs may become uncorrelated once information concerning some other r.v.(s) is known. More precisely, we may have the following independence condition:

$$P(A \mid C1, ..., Cn, U) = P(A \mid C1, ..., Cn)$$

for some collection of r.v.s U. Intuitively, we can interpret this as saying that A is determined by C1, ..., Cn regardless of U.

Combined with the chain rule, these conditional independencies allow us to replace the terms in the chain rule with the smaller conditionals. Thus, instead of explicitly keeping the joint probabilities, all we need are smaller conditional probability tables which we can then use to compute the joint probabilities. In this manner, a unique joint probability distribution is defined over the events (r.v.s) of interest. As such, we can then answer probabilistic questions of the form "What is $P(A_1 = a_1, \ldots, A_m = a_m \mid B_1 = b_1, \ldots, B_n = b_n)$?" by taking the sum and product of appropriate conditional probabilities.

In Bayesian networks, these conditional dependencies are represented as a directed acyclic graph of r.v. relationships. Directed arcs between r.v.s represent direct conditional dependencies. When all the parents of a given r.v. A are instantiated, that r.v. is said to be conditionally independent of the remaining r.v.s which are not descendants of A given it's parents. (For more details on this, see d-separation in [21].) Although Bayesian networks have been successfully used to prototype numerous intelligent systems including adversarial intent inference [26, 36] and the causal analysis tool [18, 22] for EBO, there are limitations to constructing such networks as

we have discussed above. In this paper, we recommend another uncertainty model called Bayesian Knowledge-Bases (BKBs).

Bayesian Knowledge-Bases (BKBs) are a generalization of Bayesian networks. BKBs have been extensively studied both theoretically [33, 34] and for use in knowledge engineering [25] in a wide variety of domains such as space shuttle engine diagnosis [3, 25], medical information processing [20], and freshwater aquarium maintenance [33]. BKBs provide a highly flexible and intuitive representation following a basic "if-then" structure in conjunction with probability theory. Furthermore, BKBs were designed keeping in mind typical domain incompleteness to retain semantic consistency as well as soundness of inference in the absence of complete knowledge. Bayesian networks, on the other hand, typically assume a complete probability distribution is available from the start. Also, BKBs have been shown to capture knowledge at a finer level of detail as well as knowledge that would be cyclical (hence disallowed) in BNs.

As described in [34], probabilistic models exhibiting significant local structure are common. In such models, explicit representation of that structure as done in BKBs, is advantageous, as the resulting representation is much more compact than the full table representation of the conditional probability tables (CPTs) in a BN. For example, consider the following setting: X, a binary variable, is known to be true if any of the variables Y_i is true for i from 1 to n, and X is false with probability p otherwise. The global structure here is that X depends on all the Y_i and in a BN one might represent this with a set of arcs (Y_i, X) for i from 1 to n. The representation of the distribution in the "standard" form of a CPT would require $O(2^n)$ entries. However, the (partially) given distribution also exhibits "local" structure, as when Y_i is known to be true for some i, X no longer depends on the value of Y_j for j not equal to i. The size of the representation of the conditional probabilities in terms of rules is only O(n). Although work has been done on representing local structure using other methods, such as local decision trees and default tables [35, 8], rules have significant advantages in size of the representation, as well as their better explainability.

For example, contrasting rules with decision trees as a representation of local structure, every decision tree is compactly representable as a set of rules, while the reverse is not necessarily true - the decision tree may be exponentially larger that the set of rules [2]. Although rule-based systems for representing an exact distribution exist (e.g. [23]), these systems are a (compact) notational variant of Bayesian networks, and are thus less flexible than BKBs, as they do not allow for incompleteness or cyclicity.

OVERVIEW OF ADVERSARY INTENT INFERENCING ARCHITECTURE

In this section, we provide a brief overview of our basic adversary intent architecture. In [26], we decomposed the architecture into the what/how/why model in order to provide a natural and intuitive organization of both the adversarial decision-making process and central knowledge-base. The components of our adversary intent inferencing model, and the interactions between these components, are shown in Figure 4. The three core components that comprise our architecture and functions are as follows:

1. **Goals/Foci**: Probabilistically prioritized short- and long-term goals list, representing adversary intents, objectives or foci
2. **Rationale**: A probabilistic network, representing the influences of the adversary's beliefs, both about themselves and about us, on their goals and on certain high level actions associated with those goals

3. **Actions**: A probabilistic network, representing the detailed relationships between adversary goals and the actions they are likely to perform to realize those goals

The goal component captures *what* the adversary is doing, the action component captures *how* the adversary might do it, and the rationale component infers *why* the individual is doing it. Due to the inherent uncertainty involved in adversary course of action prediction, we used Bayesian networks as the main knowledge representation for the rationale and action networks [26]. Each random variable (RV) involved in the Bayesian networks is classified into one of four classes: *axioms, beliefs, goals,* and *actions*. Each RV class is described below:

a) **Adversary axioms (X)** – represents the underlying beliefs of the adversary about themselves (vs. beliefs about our forces). This can range from an adversary's beliefs about his or her own capabilities to modeling a fanatic's belief of invulnerability. Axioms typically serve as inputs or explanations to the other RVs such as adversary goals.
b) **Adversary beliefs (B)** – represents the adversary's beliefs regarding our forces (e.g., an adversary may believe that the United States is on a crusade against them or that the United States is not carpet-bombing territory).
c) **Adversary goals (G)** – represents the goals or desired end-states of the adversary. These goals are defined as either short-term or long-term in a goals list. Further we partition goals into two types: abstract and concrete. Abstract goals have subgoals and are not immediately satisfiable by actions (e.g., preserving launchers, damage US world opinion, defeating US foreign policy). Concrete goals are satisfiable by actions.
d) **Adversary actions (A)** – represents the actions of the adversary that can typically be observed by friendly forces.

These four random variable types are arranged in the two networks: rationale network and action network. The rationale network contains all of the Belief (B), Axiom (X), and Goal (G) variables, as well as any Action (A) variables which have goals as inputs. This network is used to infer what short and long term goals the adversary may have. Once the goals are determined, the action network is used to reason on what the most likely actions will be that the adversary may carry out. The action net contains the entire set of Action (A) variables and any concrete Goal (G) variables. Figure 5 depicts a rationale network and an action network.

The AII process (as shown in Figure 4) works iteratively as follows:

1) Observables regarding the adversary such as actions and beliefs are set as evidence in both rationale and action networks (depicted as red nodes in figure). Also, feedback from analyst is set as evidence.
2) Current short- and long-term enemy foci from the foci lists are also set as evidence in both networks (depicted as green nodes).
3) The rationale network is then used to infer new goals which are set as evidence for the action network.
4) The action network is now used to predict adversarial actions.
5) The analyst is presented with the inferred goals and predicted actions.
6) The analyst provides feedback in terms of corrected goals and actions if desired.
7) The goals list is updated based on newly inferred goals and current strength of existing goals. If goals exceed a given threshold value, they are added to the list. If goals fall below a set threshold, they are removed. If goals in the short-term list persist beyond a given time threshold, they become long-term goals.
8) Go to step 1.

The inference process on both the rationale and action networks is based on belief updating [26]. In essence, given a target random variable R and evidence set E, belief updating computes P(R|E) assuming that random variables have two states (true/false) for simplicity of discussion.

As we can see in the above process, the adversary model is capable of adapting to changes in the adversaries goals and intentions over time as reflected in the enemy foci lists. Also note that there are feedback and explanation paths within the adversary intent inference (AII) model. Feedback from a human analyst, although unlikely to be totally certain, can be extremely valuable to the AII model, correcting and extending its intent inferencing logic. Explanation capabilities are essential in order for intelligence analysts, using AII, to understand why the AII model has reached particular inferences. The analysts must be able to inspect the reasoning paths used by AII so that they can develop a level of confidence in the output of the AII model.

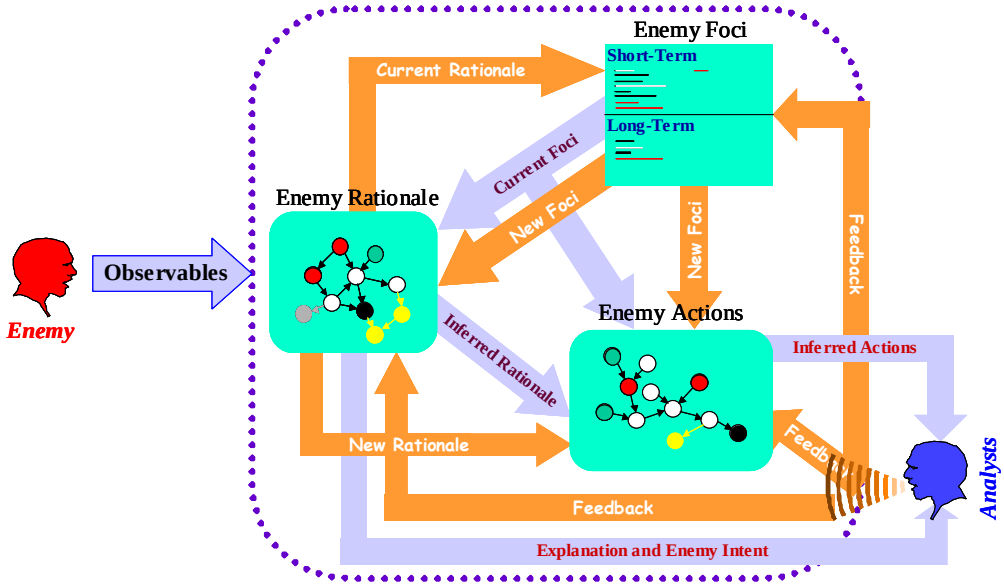


Figure 7. AII Process from [26].

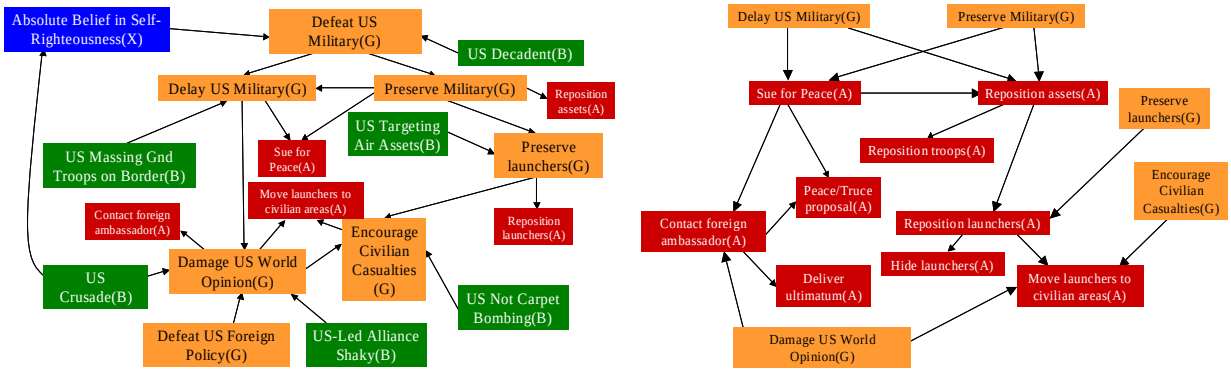


Figure 8. Rationale Network (left) and Action Network (right). The random variables are labeled according to their categories. Examples from [26].

HOW TO CONSTRUCT ADVERSARIES

In this section, we focus on how to organize the adversarial model and consider the structural relationships central to the rationale and action networks.

Semantic Structure. As we mentioned above, the four random variable types are arranged in the two networks: rationale network and action network. The rationale network contains all of the Belief (B), Axiom (X), and Goal (G) variables, as well as any Action (A) variables which have goals as inputs. This network is used to infer what short and long term goals the adversary may have. Once the goals are determined, the action network is used to reason on what the most likely actions will be that the adversary may carry out. The action net contains the entire set of Action (A) variables and any concrete Goal (G) variables. Of course, the question is "How do we systematically build such adversaries while minimizing the bottleneck-effect of knowledge acquisition?"

In [26, 36], we proposed and utilized a fundamental structure of how the networks should be organized and built. Here, we expand the semantic structure in order to further account for the impact of blue actions on the adversary's actions. As a rule, belief variables are independent and serve as inputs to Axioms, Goals, or Actions. Belief variables can be categorized into two basic types: *strategic beliefs* and *tactical beliefs*. Strategic beliefs include philosophy, strategic goals, and general characteristics/behaviors of blue from the adversary's point of view (such as those described in the earlier examples). Tactical beliefs represent actionable blue events such as physical repositioning of assets, specific kinetic attacks, etc. While it seems reasonable to construct a dependency structure among the belief variables especially tactical beliefs that represent sequences or hierarchies of blue actions, this increases the complexity of the networks. Since blue actions are typically known with certainty, the belief variables can be set as evidence which has the effect of rendering the beliefs independent. Still, if the added complexity is acceptable, partial evidence on blue activity can reflect the "fog of war" in which the tactical beliefs can be probabilistically inter-related and thus provides a natural sensitivity/what-if analysis of blue actions with respect to the adversary. In this case, we would amend the semantic structure that organizes the beliefs into (multiple) hierarchies of beliefs such that Belief variables can serve as input to other Belief variables, strategic beliefs serve as input to Goals, and tactical beliefs can serve as input to Goals and Actions. We also allow beliefs to occur in either the rationale or action network.

Next, Axioms have strategic Beliefs as inputs and serve as inputs to Goals and other Axioms. Goals have Axioms and Beliefs as inputs and serve as inputs to Actions or other Goals. Actions have Goals and tactical Beliefs as inputs and can only be inputs to other Actions. Basically, the structure follows an intuitive hierarchical pre- and post-condition organization. Hence, there is a natural dominance relationship between related variable types, say Axioms that are descendants of other Axioms. This dominance reflects the fact that one variable may be "more general", "more abstract", "aggregates", "pre-conditions", etc. with respect to a descendant variable.

In order to maintain the appropriate division of variables among the rationale and action networks, Goal variables are partitioned into abstract goals and concrete goals. Abstract goals are goals composed of additional Goal variables. Concrete goals are goals that are immediately actionable, i.e., satisfied by one or more Action variables. As such, abstract goals can only appear in the rationale network and are critical to providing the proper explanations for adversary rationale. Concrete goals must appear in both networks and serves as the causal "glue" between the networks.

Finally, there are three basic rules that maintains the semantic integrity of the AII:

1. All axioms, beliefs, goals and actions occur in at least one of the adversary rationale or action networks.
2. Given a concrete goal G, if A is an action node with input G, then G, A, and the inputs of A must occur in both networks with the same connection structure.

3. All Bayesian networks must be directed acyclic graphs (DAGs).

Rule 2 is particularly important in that it permits the propagation of reasoning between the two networks. Such propagation occurs in both directions: rationale to action to rationale, reflecting the predictive and explanatory processes in the AII.

Causal Structuring. While the rules above for structuring the relationships between the major classes of variables ensures a proper semantic organization throughout the AII model, there still remains basic engineering issues in constructing the specific relationships especially with regards to causal structuring. Consider scenarios where you have collections of potentially mutually exclusive events. For example,

- Case 1: Simple unit movement – Ground Unit A can move in one of 8 possible directions {N, NE, E, SE, S, SW, W, NW}. If the domain consists of a fixed octagonal-grid, then the movement direction (not moving is not considered here) must be mutually exclusive. Often, a single random variable (rv) is used in this case to indicate direction. However, if fidelity of movement needs to be considered, then one can have a N NE movement. As such, eight rvs corresponding to the eight directions of movement for a given unit are used where each rv has a true/false value. Hence, N NE is represented by two true assignments. This case is not recommended unless absolutely necessary given the problems of a potential N S setting with the rvs. [Note – there is a solution to handling this latter problem by generalizing our mutual exclusion mechanism as described later. In essence, to prevent a simultaneous N S setting, one can introduce a new rv that causes such a N S setting to have a 0 probability.]
- Case 2: Complimentary actions – Air Unit A attacking SAM Site S and Air Unit B attacking SAM Site S. If only one Air Unit can be engaging a SAM Site at any given time, then these two events must be mutually exclusive. A typical approach would be to have a single rv that has states {Air Unit A attacking SAM Site S, Air Unit B attacking SAM Site S, no attack against SAM Site S}. Let's complicate the problem by considering that Air Unit A can also attack Bunker Q. If the events are mutually exclusive, we must make sure that if the rv controlling attacks on SAM Site S must make sure that it is a parent of the rv to stop Air Unit A from attacking Bunker Q if Air Unit A must also attack SAM Site S.

The largest knowledge acquisition difficulty with the examples above when one uses single rvs to deal with mutual exclusion is the problem of capturing the appropriate interactions between all rvs such as in Case 2. Such interactions can lead to serious looping and spaghetti-like constructs which ultimately leads to a failure in capturing the correct information. Furthermore, we incur a significant space and time complexity because of the explosion in the size of the conditional probability tables when a single rv is used to capture mutual exclusion since that rv has to be the parent of any rv that needs information regarding one of its states. For example, if some rv wants to know Air Unit A's current attack, it must look at the rv which has at least 3 states.

A simple solution to mutual exclusion in our model is the following: Assume all rvs are true/false only. Let A1, A2, …, An be rvs that need to be mutually exclusive. Construct a new rv B that is also true/false with parents A1, …, An. Now, set the conditional probability table for B as follows: P(B=true| only one Ai is true) = 1.0, 0.0 otherwise. P(B=false|…) = 1.0 – P(B=true|…), of course. [Note – we assume here that one of the Ai's must be true. This can be easily modified if it is necessary to model when all Ai's are false as a valid state.] In order for the model to function properly, make B of type Axiom. If mutual exclusion is desired between A1, …, An, then set B=true as evidence. This will guarantee that only one of the rvs will be set to true since all other scenarios will result in a 0 probability. By separating out the states, this will reduce the explosion and spaghetti structure involved in the single rv method. Construction should also be more methodical.

We also recommend the migration to Bayesian Knowledge-Bases as the knowledge representation [33, 34]. BKBs do not require a complete instantiation of all conditional probabilities since BKBs are rule-based in nature. Consider a BKB rule: If A = true and B = true then C = false with probability p. As such, p = P(C=false | A=true, B=true). We do not have to specify other combinations of A, B, and C unless we wish to or the information is relevant. Also, we can capture: If B=false and D=true and E=false, then C=true with probability p' can also be handled in a BKB together with the last rule. Finally, in situations where we find that we have a rule: If C = true, then A=false with probability p'. We can also capture this in a BKB. Thus, a BKB allows for incomplete information (avoids explosion from not having to specify all conditional probabilities), content-specific rules, and limited forms of cyclicity. As presented in BKBs, there exists a simple set of rules for building BKB rules that guarantess the BKB is consistent. The recommended mutual exclusion mechanisms above can be directly used in BKBs but without the 2^n explosion since we would only need to encode the probabilities with a non-zero value. Thus, only need n rules. As a starting recommendation, try incompleteness and context-specific rules before trying cyclicity.

TEMPLATES FOR BUILDING AIIS

In this section, we describe a tool we have developed to assist in building the AII. In particular, the tool provides the functionality for building the various random variables, action network, and rationale network for the AII. Furthermore, it provides the capability for generating templates and instantiating them for the AII that should greatly ease the construction process. With such a library of templates, this will allow the AII's to be modified "on-demand." For example, assume the current AII is modeling an air commander adversary and his/her assets and capabilities. If a new asset is discovered such as an additional air base, a new SAM site, etc., this capability can be directly incorporated into the AII model by identifying the relevant template that defines these capabilities in general and appropriately instantiating them with regards to the specific characteristics and behavioral probabilities of the new capability, etc.

The Adversary Intent Inference model editor, also known as the AII Template Generator (ATG), is a tool for quickly constructing probabilistic networks specific to AII. This tool is being built because the creation of a set of Bayesian Network files required by the Adversary Intent Inference system is very tedious and time consuming. Prior to ATG, there have been compatibility problems between different applications that help build BNs and the rule/constraint handling that is central to the AII described earlier. The ATG allows for compatibility between different file types, such as from different Bayesian Network editors, by providing an import and export feature to translate existing BNs between applications. The ATG also provides validation of current BNs against the AII semantic structure rules. The BNs used in AII are very specific as well as structurally constrained by the types of random variables used. Hence error handling, prior to ATG, was left mostly to the creator of the files and to their respective intuition. Consequently, BN files created for the use in the AII take longer to build and are more likely to contain errors. The main feature of the ATG is the ability to create BN nodes with their respective random variables based on template information, which is user specified and can be used globally between different AII systems/projects. This allows for quick generation of random variables, nodes, and default probabilities for the nodes in the BNs.

The ATG allows the easy implementation of the same BN across different AII applications/systems. The ATG contains import and export features which actually converts other formats into its own XML format. The import features lets the application import, into an existing project, files from a number of BN application formats. Because there are three different specific files loaded into the Adversary Intent Inference system: the AII model (*.am), the

rationale Bayesian Network (*.gra), and the action Bayesian Network (*.gra), the importing of an existing network will be in a similar format, with the advantage of generating the random variables automatically. Similarly the export feature will generate from an existing project the three files required by the Adversary Intent Inference system.

Template models offer quick and easy creation of BNs and random variables that comply with their respective constraints. Templates should be used for repetitive tasks that will need nodes of the same type of random variables as inputs (parents), or will need nodes with similar types of input, but with pre-set probability values. Template models are stored in the application globally; therefore, more than one project can then use the same template(s). Templates are organized into groups. The purpose of a template is to generate a specific model with pre-set probabilities in which random variables are bound to template variables that are of a specific type (Axiom | Goal | Action | Belief). Figure 9 shows the template dialog and the results after applying the template onto an empty Bayesian Network.

Figure 10 graphically depicts the Template stored in ATG, where a specific parent node (Go_IraquiMassForces of type Goal) is used as a parent onto a variable child node (Ac_IraquiForcesMassing), and the variable child node can be any node of type Action. Specifying the probabilities for the child node such as AtAreaA, AtAreaB, and AtAreaC will let the template tool generate the three nodes of type Action (Ac_IraquiForcesMassingAtAreaA, Ac_IraquiForcesMassingAtAreaB, and Ac_IraquiForcesMassingAtAreaC) with their respective parent node and pre-set probabilities after applying the template, thus generating quickly the set of nodes from the template used.

There are two ways that templates can be used in the ATG application:

1. Templates can be applied to existing or newly created models (nodes). This adds and maintains consistency within the model by only allowing the random variables of specific type to be bounded within the model, conforming to the template specification.

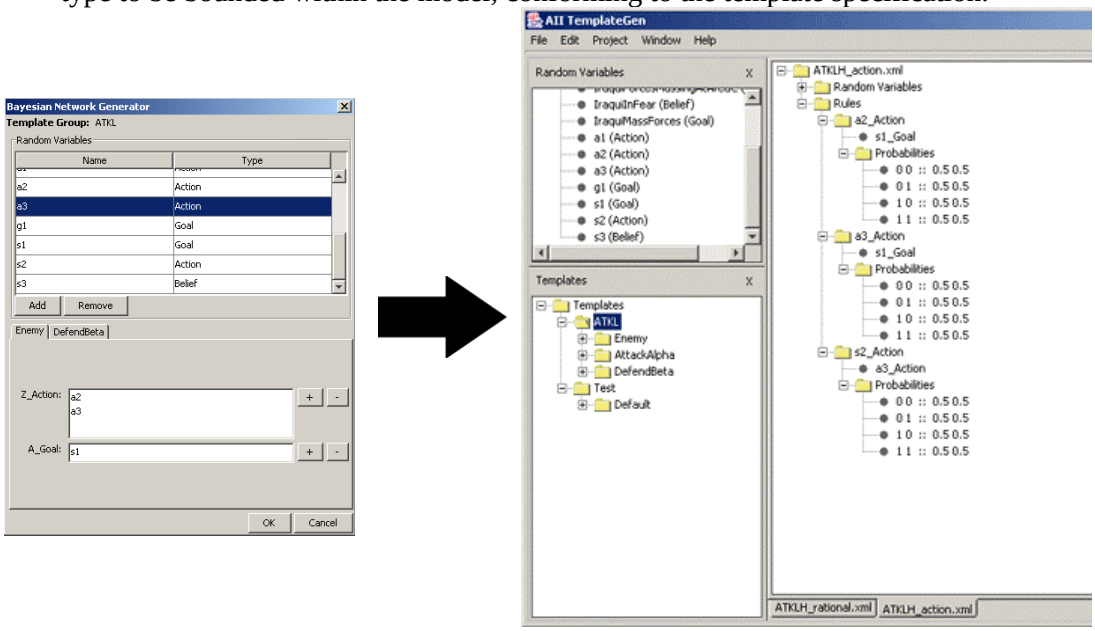


Figure 9. Instantiating a BN from a template.

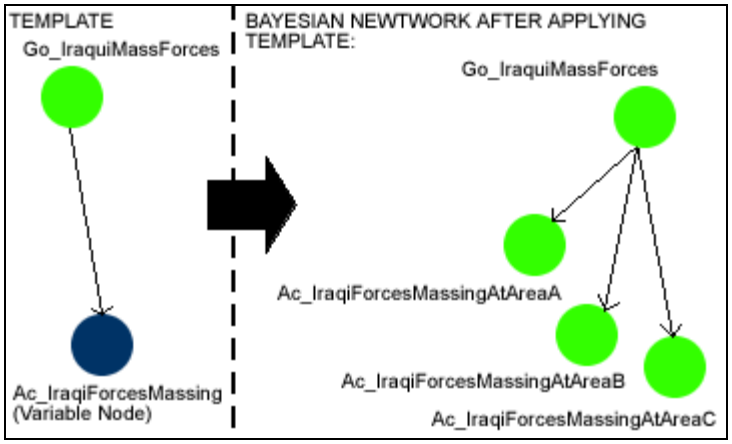


Figure 10. Graphical representation of a template and its instantiation.

2. Templates and template groups can be dragged & dropped onto the BNs. This option is more complex, but allows us to systematically and consistently create new nodes. Only the templates supported by the rule for the type of BN are enabled, so that models that do not comply with the rule are not created. Next, the existing random variables are listed, but new random variables can be added as well, so that binding can be applied onto each of the templates' inputs (the nodes parents) and outputs (the generated node). The systematic binding will allow as many different combinations for outputs, creating many different instances of the model (node) with distinct output values. Every node is an output, where there are different outputs/nodes with the same parents and probabilities, but there cannot exist any duplicates of one node with different parents. See Figure 11.

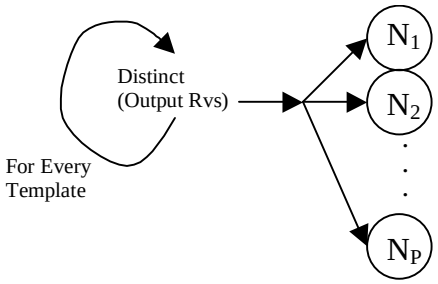


Figure 11. Combinations of random variables are used to instantiate the templates.

The ATG is a complementary tool to the AII system. The ATG allows quick creation of a set of BNs to use in the AII system. By allowing import and export of BN file formats ATG can be used as a "bridge" between different applications. Also, through constraint checking and validation, the development of these BNs to the specification of the AII rules should make debugging and troubleshooting much simpler. The use of templates in the development of these networks can reduce the creation/development time and helps avoid some repetitive tasks that can be done in a systematic manner. A library of such templates can also be exploited computationally to construct/update AII components on-the-fly during execution of the AII.

The next step for the ATG is to improve load efficiency on projects, allow undo procedures, improve rule and constraint checking of BN to increase accuracy, extend import and export capabilities, and implement a graphical view of the BNs by displaying a directed graph of the parent-to-child node relationships. We are also migrating towards a Bayesian Knowledge-Base representation for ATG.

ABOUT DECEPTION

As we described earlier, our model of the adversary's biases and fundamental beliefs are modeled with the variables in the adversary Axioms. Again, this is different from those in adversary Beliefs which represent the adversary's belief about us – blue forces. To recap, the current enemy foci/goals and the observables regarding the adversary are set as evidence in order to predict future adversary actions, determine adversary goals and changes in foci, etc. In essence, the collection of evidence represents the adversary's *perceptions*. Clearly, the perceptions which are set as evidence (observables) heavily impacts the predicted adversary goals and actions and serves as the basis for explaining the predictions. However, those perceptions which have not been set as evidence because they have yet to be observed play a very important factor in adversary analysis. The explanation of a predicted goal or action may rest on assumptions regarding the unobserved perceptions. This leads to a rich series of what-if analyses that can be readily conducted to better support the model's predictions as well as direct blue forces to gather more "targeted" information in order to support or refute a given prediction.

By considering adversary perceptions within the AII, this naturally provides a mechanism for handling deception (both red and blue). The objective of a deception can range from masking the true intent/goal of red or blue by hiding actions (say, camouflage or diverting attention from a given action say in urban operations [17]) to influencing and/or changing red or blue goals and desired end-states through misdirection or misinformation [29]. Combined with the what-if analysis and explanatory capabilities of AII, we can analyze the effects of deceptions as follows: For blue deceptions, we can naturally analyze their impacts by setting various combinations of adversary perceptions as observables corresponding to the target deceptions. The predictions and explanations generated can then be used to evaluate the effectiveness of the blue deception especially if the goal of the deception was to alter red's actions or goals. Similarly, to detect/determine red deceptions, various observables can be applied and sensitivity analysis can be conducted on the resulting predictions to determine potential red COAs. From the sensitivity analysis, this can identify to blue what additional reconnaissance information should be obtained in order to pinpoint the red deception and red's true goals. As we can see, the AII cognitive architecture provides a natural interface to take into account deception.

Acknowledgements. This work was supported in part by Air Force Research Labs, Information Directorate, Grant No. F30602-01-1-0595. Special thanks to Robert Hillman and Jim Hanna for insights and encouragements on this effort. Also, thanks to Frank Vetesi (LM ATL) for his practical comments on constructing adversaries. Finally, I would also like to especially thank John Graniero (AFRL, Information Institute), Don Monk (AFRL, Human Effectiveness), and Scott Brown (USAF/ESC) for their tremendous support for this research project.

REFERENCES

2. Aizenstein, Howard, Blum, Avrim, Khardon, Roni, Kushilevitz, Eyal, Pitt, Leonard, and Roth, Dan, "On Learning Read-k -Satisfy-j DNF," *SIAM Journal on Computing* **27(6)**, 1515-1530, 1998.

3. Banks, Darwyn O, "Acquiring Consistent Knowledge for Bayesian Forests," MS thesis, AFIT/GSO/ENG/95M-01. Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, March 1995.
4. Behler, R., "Homeland Information: AOC Can Coordinate U.S. Terror Defense," *Defense News* **13**, 2001.
5. Bell, Benjamin and Santos, Eugene, Jr., (Eds.), *Intent Inference for Collaborative Tasks: Papers from the 2001 Fall Symposium*, AAAI Press, Menlo Park, CA, 2001.
6. Bell, Benjamin, Santos, Eugene, Jr., and Brown, Scott M., "Making Adversary Decision Modeling Tractable with Intent Inference and Information Fusion," *Proceedings of the 11th Conference on Computer Generated Forces and Behavioral Representation*, 535-542, Orlando, FL, 2002.
7. Bigelow, James H. and Davis, Paul K., "Implications for Model Validation of Multiresolution, Multiperspective Modeling (MRMPM) and Exploratory Analysis," MR-1750, RAND, 2003.
8. Boutilier, Craig, Friedman, Nir, Goldszmidt, Moises, and Koller, Daphne, "Context-Specific Independence in Bayesian Networks," *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 1996.
9. Brown, Scott M. and Santos, Eugene, Jr., "Active User Interfaces for Building Decision-Theoretic Systems," *Proceedings of the 1st Asia-Pacific Conference on Intelligent Agent Technology*, 244-253, Hong Kong, 1999.
10. Brown, Scott M., Santos, Eugene, Jr., and Banks, Sheila B., "Utility theory-based user models for intelligent interface agents," *Lecture Notes in Artificial Intelligence 1418: Advances in Artificial Intelligence - AI '98*, 378-392, Springer-Verlag, 1998.
11. Brown, Scott M., Santos, Eugene, Jr., Banks, Sheila B., and Oxley, Mark, "Using Explicit Requirements and Metrics for Interface Agent User Model Correction," *Proceedings of the Second International Conference on Autonomous Agents*, 1-7, Minneapolis/St. Paul, MN, 1998.
12. Brown, Scott M., Santos, Eugene, Jr., and Bell, Benjamin, "Knowledge Acquisition for Adversary Course of Action Prediction Models," *Proceedings of the AAAI 2002 Fall Symposium on Intent Inference for Users, Teams, and Adversaries*, Boston, MA, 2002.
13. Davis, Paul, "Synthetic Cognitive Modeling of Adversaries for Effects-Based Planning," *Proceedings of the SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, Orlando, FL, 2003
14. Fayette, D. F., "Effects-Based Operations: Application of new concepts, tactics, and software tools support the Air Force vision for effects-based operations", Air Force Research Laboratory Technology Horizons, IF-00-15, 2001.
15. Franke, J., Brown, S. M., Bell, B., and Mendenhall, H., "Enhancing Teamwork Through Team-Level Intent Inference," *Proceedings of the International Conference on Artificial Intelligence (IC AI 2000)*, Las Vegas, NV, 2000.
16. Geddes, Norm, "The Use of Individual Differences in Inferring Human Operator Intentions," *Proceedings of the Second Annual Aerospace Applications of Artificial Intelligence Conference*, 1986.
17. Gerwehr, Scott and Glenn, Russell W., "Unweaving the Web: Deception and Adaptation in Future Urban Operations," MR-1495, RAND Arroyo Center, 2002.
18. Gossink, Don E. and Lemmer, John F., "Recursive Noisy OR – A rule for estimating complex probabilistic causal interactions," *IEEE Transactions of Systems, Man, and Cybernetics: Part B*, to appear.
19. McCrabb, M. and Caroli, J., "Behavioral Modeling And War Gaming For Effects-Based Operations", *Proceedings Of The Analyzing Effects Based Operations Workshop, Military Operations Research Society*, Vienna, VA, January 2002.
20. Nguyen, Hien, Saba, G. Mitchell, Santos, Eugene, Jr., and Brown, Scott M., "Active User Interface in a Knowledge Discovery and Retrieval System," *Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI 2000)*, Las Vegas, NV, 2000.
21. Pearl, Judea, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, 1988.
22. Phister, Paul W. Jr. and Plonisch, Igor G. "Military Applications of Information Technologies," *Air & Space Power Journal* **18(1)**, 77-90, 2004.
23. Poole, David, "Probabilistic Horn Abduction and Bayesian Networks," *Artificial Intelligence* **64(1)**, 81-129, 1993.

24. Revello, Timothy, McCartney, Robert, and Santos, Eugene, Jr., "Multiple Strategy Generation for War Gaming," *Proceedings of the SPIE Defense & Security Symposium*, Orlando, FL, 2004.
25. Santos, Eugene, Jr., "Verification and Validation of Knowledge-Bases Under Uncertainty," *Data and Knowledge Engineering* **37**, 307-329, 2001.
26. Santos, Eugene, Jr., "A Cognitive Architecture for Adversary Intent Inferencing: Knowledge Structure and Computation," *Proceedings of the SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, 182-193, Orlando, FL, 2003.
27. Santos, Eugene, Jr. and Bell, Benjamin (Eds.), *Intent Inference for Users, Teams, and Adversaries: Papers from the 2002 AAAI Fall Symposium*, AAAI Press, Menlo Park, CA, 2003.
28. Santos, Eugene, Jr., Brown, Scott M., Lejter, Moises, Ngai, Grace, Banks, Sheila B., and Stytz, Martin R., "Dynamic User Model Construction with Bayesian Networks for Intelligent Information Queries," *Proceedings of the 12th International FLAIRS Conference*, 3-7, Orlando, FL, 1999.
29. Santos, Eugene, Jr. and Johnson, Gregory, "Toward Detecting Deception in Intelligent Systems," *Proceedings of the SPIE Defense & Security Symposium*, Orlando, FL, 2004.
30. Santos, Eugene, Jr., Nguyen, Hien, and Brown, Scott M., "Kavanah: An Active User Interface Information Retrieval Application," *Proceedings of the 2nd Asia-Pacific Conference on Intelligent Agent Technology*, 412-423, 2001.
31. Santos, Eugene, Jr., Nguyen, Hien, Zhao, Qunhua, and Pukinskis, Erik, "Empirical Evaluation of Adaptive User Modeling in a Medical Information Retrieval Application," *Lecture Notes in Artificial Intelligence 2702: User Modeling 2003* (Eds. P. Brusilovsky, A. Corbett, and F. de Rosis), 292-296, Springer, Johnstown, PA, 2003.
32. Santos, Eugene, Jr., Nguyen, Hien, Zhao, Qunhua, and Wang, Hua, "User Modelling for Intent Prediction in Information Analysis," *Proceedings of the 47th Annual Meeting for the Human Factors and Ergonomics Society (HFES-03)*, 1034-1038, Denver, CO, 2003.
33. Santos, Eugene, Jr. and Santos, Eugene S., "A Framework for Building Knowledge-Bases Under Uncertainty," *Journal of Experimental and Theoretical Artificial Intelligence* **11**, 265-286, 1999.
34. Santos, Eugene, Jr., Santos, Eugene S., and Shimony, Solomon Eyal, "Implicitly Preserving Semantics During Incremental Knowledge Base Acquisition Under Uncertainty," *International Journal of Approximate Reasoning* **33(1)**, 71-94, 2003.
35. Shimony, Solomon Eyal, "The Role of Relevance in Explanation I: Irrelevance as Statistical Independence," *International Journal of Approximate Reasoning* **8(4)**, 281-324, 1993.
36. Surman, Joshua, Hillman, Robert, and Santos, Eugene, Jr., "Adversarial Inferencing for Generating Dynamic Adversary Behavior," *Proceedings of the SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls: AeroSense 2003*, 194-201, Orlando, FL, 2003.